CARBayes version 6.1.1: An R Package for Spatial Areal Unit Modelling with Conditional Autoregressive Priors

Duncan Lee
University of Glasgow

Abstract

This is a vignette for the R package **CARBayes** version 6.1.1, and is an updated version of a paper in the Journal of Statistical Software in 2013 Volume 55 Issue 13 by the same author. The package implements univariate and multivariate spatial generalised linear mixed models for areal unit data, with inference in a Bayesian setting using Markov chain Monte Carlo (MCMC) simulation using a single or multiple Markov chains.. The response variable can be binomial, Gaussian, multinomial, Poisson or zero-inflated Poisson (ZIP), and spatial autocorrelation is modelled by a set of random effects that are assigned a conditional autoregressive (CAR) prior distribution. A number of different models are available for univariate spatial data, including models with no random effects as well as random effects modelled by different types of CAR prior. Additionally, a multivariate CAR (MCAR) model for multivariate spatial data is available, as is a two-level hierarchical model for modelling data relating to individuals within areas. The initial creation of this package was supported by the Economic and Social Research Council (ESRC) grant RES-000-22-4256, and on-going development has been supported by the Engineering and Physical Science Research Council (EPSRC) grant EP/J017442/1, ESRC grant ES/K006460/1, Innovate UK / Natural Environment Research Council (NERC) grant NE/N007352/1, and the TB Alliance. The new functionality in version 6.0 is that 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, R package.

1. Introduction

Data relating to a set of non-overlapping spatial areal units are prevalent in many fields, including agriculture (Besag and Higdon 1999), ecology (Brewer and Nolan 2007), education (Wall 2004), epidemiology (Lee 2011) and image analysis (Gavin and Jennison 1997). There are numerous motivations for modelling such data, including ecological regression (see Wakefield 2007 and Lee *et al.* 2009), disease mapping (see Green and Richardson 2002 and Lee 2011) and Wombling (see Lu *et al.* 2007, Ma and Carlin 2007). The set of areal units on which data are recorded can form a regular lattice or differ largely in both shape and size, with examples of the latter including the set of electoral wards or census tracts corresponding to a city or country. In either case such data typically exhibit spatial autocorrelation, with

observations from areal units close together tending to have similar values. A proportion of this spatial autocorrelation may be modelled by known covariate risk factors in a regression model, but it is common for spatial structure to remain in the residuals after accounting for these covariate effects. This residual spatial autocorrelation can be induced by unmeasured confounding, neighbourhood effects, and grouping effects.

The most common remedy for this residual autocorrelation is to augment the linear predictor with a set of spatially autocorrelated random effects, as part of a Bayesian hierarchical model. These random effects are typically represented with a conditional autoregressive (CAR, Besag *et al.* 1991) prior, which induces spatial autocorrelation through the adjacency structure of the areal units. A number of CAR priors have been proposed in the literature, including the intrinsic and Besag-York-Mollié (BYM) models (both Besag *et al.* 1991), as well as the alternative developed by Leroux *et al.* (2000).

However, these CAR priors force the random effects to exhibit a single global level of spatial autocorrelation, ranging from independence through to strong spatial smoothness. Such a uniform level of spatial autocorrelation for the entire region maybe unrealistic for real data, which instead may exhibit sub-regions of spatial autocorrelation separated by discontinuities. A number of approaches have been proposed for extending the class of CAR priors to deal with localised spatial smoothing amongst the random effects, including Lee and Mitchell (2012), and Lee and Sarran (2015).

The models described above are typically implemented in a Bayesian setting, where inference is based on Markov chain Monte Carlo (MCMC) simulation. The most commonly used software to implement this class of models is the BUGS project (Lunn *et al.* 2009, WinBUGS and OpenBUGS), which has in-built functions to implement the intrinsic and BYM models. CAR models can also be implemented in R using Integrated Nested Laplace Approximations (INLA, *http://www.r-inla.org/*), using the package **INLA** (Rue *et al.* 2009).

CARBayes (Lee 2013) is the premier R package for modelling spatial areal unit data with conditional autoregressive type spatial autocorrelation structures in a Bayesian setting using MCMC simulation. Its main advantages are firstly ease of use because: (1) the spatial adjacency information is easy to specify as a neighbourhood (adjacency) matrix; and (2) given the neighbourhood matrix, models can be implemented by a single function call. Secondly, **CARBayes** can implement a much wider class of spatial areal unit models than say WinBUGS, and the univariate or multivariate response data can follow binomial, Gaussian, multinomial, Poisson or zero-inflated Poisson (ZIP) distributions, while a range of CAR priors can be specified for the random effects. Additionally, a two-level hierarchical model is available for modelling data relating to individuals within areas. Spatio-temporal models for areal unit data using CAR type priors can be implemented using the sister package **CARBayesST** (Lee *et al.* 2018).

The aim of this vignette is to present **CARBayes**, by outlining the class of models that it can implement and illustrating its use by means of 2 worked examples. Section 2 outlines the general Bayesian hierarchical model that can be implemented in the **CARBayes** package, while Section 3 details the inputs and outputs for the software. Sections 4 to 5 give two worked examples of using the software, including how to create the neighbourhood matrix and produce spatial maps of the results. Finally, Section 7 contains a concluding discussion, and outlines areas for future development.

2. Spatial models for areal unit data

This section outlines the class of spatial generalised linear mixed models for areal unit data that can be implemented in **CARBayes**. Inference for all models is set in a Bayesian framework using MCMC simulation. The majority of the models in **CARBayes** relate to univariate spatial data and are described in Section 2.1, while models for multivariate spatial data and two-level data relating to individuals within areas are described in Sections 2.2 and 2.3.

2.1. Univariate spatial data models

The study region $\mathcal{S}$ is partitioned into K non-overlapping areal units $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_K\}$, which are linked to a corresponding set of responses $\mathbf{Y} = (Y_1, \ldots, Y_K)$, and a vector of known offsets $\mathbf{O} = (O_1, \ldots, O_K)$. Missing, `NA`, values are allowed in the response $\mathbf{Y}$ except for the `S.CARlocalised()` function, which does not allow them due to model complexity and corresponding poor predictive performance. These missing values are treated as additional unknown parameters, and are updated in the MCMC algorithm using a data augmentation approach Tanner and Wong (1987). The spatial variation in the response is modelled by a matrix of covariates $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_K)$ and a spatial structure component $\boldsymbol{\psi} = (\psi_1, \ldots, \psi_K)$, the latter of which is included to model any spatial autocorrelation that remains in the data after the covariate effects have been accounted for. The vector of covariates for areal unit $\mathcal{S}_k$ are denoted by $\mathbf{x}_k = (1, x_{k1}, \ldots, x_{kp})$, the first of which corresponds to an intercept term. The general spatial generalised linear mixed model is given by

$$\begin{aligned} Y_k|\mu_k &\sim f(y_k|\mu_k, \nu^2) \quad \text{for } k = 1, \ldots, K \\ g(\mu_k) &= \mathbf{x}_k^\top \boldsymbol{\beta} + O_k + \psi_k \\ \boldsymbol{\beta} &\sim \mathrm{N}(\boldsymbol{\mu}_\beta \boldsymbol{\Sigma}_\beta) \\ \nu^2 &\sim \text{Inverse-Gamma}(a, b). \end{aligned} \tag{1}$$

The expected value of Y_k is denoted by $\mathsf{E}(Y_k) = \mu_k$, while ν^2 is an additional scale parameter that is required if the Gaussian family is used. The latter is assigned a conjugate inverse-gamma prior distribution, where the default specification is $\nu^2 \sim \text{Inverse-Gamma}(1, 0.01)$. The vector of regression parameters are denoted by $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$, and non-linear covariate effects can be incorporated into the above model by including natural cubic spline or polynomial basis functions of the covariates in $\mathbf{X}$. A multivariate Gaussian prior is assumed for $\boldsymbol{\beta}$, and the mean $\boldsymbol{\mu}_\beta$ and diagonal variance matrix $\boldsymbol{\Sigma}_\beta$ can be chosen by the user. Default values specified by the software are a constant zero-mean vector and diagonal elements of $\boldsymbol{\Sigma}_\beta$ equal to 100,000. The expected values of the responses are related to the linear predictor via an invertible link function $g(.)$, and **CARBayes** can fit the following data likelihood models:

- **Binomial -** $Y_k \sim \text{Binomial}(n_k, \theta_k)$ and $\ln(\theta_k/(1-\theta_k)) = \mathbf{x}_k^\top \boldsymbol{\beta} + O_k + \psi_k$.
- **Gaussian -** $Y_k \sim \mathrm{N}(\mu_k, \nu^2)$ and $\mu_k = \mathbf{x}_k^\top \boldsymbol{\beta} + O_k + \psi_k$.
- **Poisson -** $Y_k \sim \text{Poisson}(\mu_k)$ and $\ln(\mu_k) = \mathbf{x}_k^\top \boldsymbol{\beta} + O_k + \psi_k$.
- **ZIP -** $Y_k \sim \text{ZIP}(\mu_k, \omega_k)$. The zero-inflated Poisson model is used to represent data containing an excess of zeros, and is a mixture of a point mass distribution based at

Model	Equation	Description
`S.glm()`		fits a model with no random effects and thus is a generalised linear model. This model can be implemented with binomial, Gaussian, Poisson and ZIP data likelihoods.
`S.CARbym()`	(2)	fits the convolution or Besag-York-Mollie (BYM) CAR model outlined in Besag *et al.* (1991). This model can be implemented with binomial, Poisson and zip data likelihoods.
`S.CARleroux()`	(3)	fits the CAR model proposed by Leroux *et al.* (2000). This model can also fit the intrinsic CAR model proposed by Besag *et al.* (1991), as well as a model with independent random effects. This model can be implemented with binomial, Gaussian, Poisson and ZIP data likelihoods.
`S.CARdissimilarity()`	(5) - (6)	fits the localised spatial autocorrelation model proposed by Lee and Mitchell (2012). This model can be implemented with binomial, Gaussian and Poisson data likelihoods.
`S.CARlocalised()`	(7)	fits the localised spatial autocorrelation model proposed by Lee and Sarran (2015). This model can be implemented with binomial and Poisson data likelihoods.

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

zero and a Poisson distribution with mean μ_k. The probability that observation Y_k is in the point mass distribution based at zero (called a structural zero) is ω_k, and (μ_k, ω_k) are modelled by

$$\ln(\mu_k) = \mathbf{x}_k^\top \boldsymbol{\beta} + O_k + \psi_k \qquad \ln\left(\frac{\omega_k}{1-\omega_k}\right) = \mathbf{v}_k^\top \boldsymbol{\delta} + O_k^{(2)}.$$

Here $(\mathbf{v}_k, O_k^{(2)})$ are respectively covariates and an offset term that determine the probability that observation Y_k is in the point mass distribution, while $\boldsymbol{\delta}$ are the corresponding regression parameters. In implementing the model a binary random variable Z_k is sampled for each observation Y_k such that $Y_k = 0$, where $Z_k = 1$ if Y_k comes from the point mass distribution, and $Z_k = 0$ if Y_k comes from the Poisson distribution. Further details about ZIP models are given by Ugarte *et al.* (2004).

In the binomial model above n_k is the number of trials in the kth area, while θ_k is the probability of success in a single trial. **CARBayes** can implement a number of different spatial random effects models for $\boldsymbol{\psi}$, and they are summarised in Table 1. In most cases the spatial structure component $\boldsymbol{\psi}$ includes a set of random effects $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_K)$, which come from a conditional autoregressive model. These models can be written in the general form $\boldsymbol{\phi} \sim \mathrm{N}(\mathbf{0}, \tau^2 \mathbf{Q}(\mathbf{W})^{-1})$, where $\mathbf{Q}(\mathbf{W})$ is the precision matrix that may be singular (e.g. the intrinsic model). This matrix controls the spatial autocorrelation structure of the random effects, and is based on a non-negative symmetric $K \times K$ neighbourhood (or adjacency) matrix $\mathbf{W}$. The kjth element of the neighbourhood matrix w_{kj} represents the spatial closeness between areas $(\mathcal{S}_k, \mathcal{S}_j)$, with positive values denoting geographical closeness and zero values denoting non-closeness. Additionally, diagonal elements $w_{kk} = 0$.

A binary specification for $\mathbf{W}$ based on geographical contiguity is most commonly used, where $w_{kj} = 1$ if areal units $(\mathcal{S}_k, \mathcal{S}_j)$ share a common border (denoted $k \sim j$), and is zero otherwise. This specification forces (ϕ_k, ϕ_j) relating to geographically adjacent areas (that is where $w_{kj} = 1$) to be autocorrelated, whereas random effects relating to non-contiguous areal units (that is where $w_{kj} = 0$) are conditionally independent given the values of the remaining random effects. A binary specification is not necessary in **CARBayes** except for the function `S.CARdissimilarity()`, as the only requirement is that $\mathbf{W}$ is non-negative and symmetric. However, each area must have at least one positive element $\{w_{kj}\}$, meaning the row sums of $\mathbf{W}$ must be positive. CAR priors are commonly specified as a set of K univariate full conditional distributions $f(\phi_k|\boldsymbol{\phi}_{-k})$ for $k = 1, \ldots, K$, where $\boldsymbol{\phi}_{-k} = (\phi_1, \ldots, \phi_{k-1}, \phi_{k+1}, \ldots, \phi_K)$, which is how they are presented below. We now outline the five models that **CARBayes** can fit.

A model with no random effects

`S.glm()`

The simplest model that **CARBayes** can implement is a generalised linear model, which is based on (1) with the simplification that $\psi_k = 0$ for all areas k.

Globally smooth CAR models

`S.CARbym()`

The convolution or Besag-York-Mollie (BYM) CAR model outlined in Besag *et al.* (1991) contains both spatially autocorrelated and independent random effects and is given by

$$\begin{aligned} \psi_k &= \phi_k + \theta_k \\ \phi_k|\boldsymbol{\phi}_{-k}, \mathbf{W}, \tau^2 &\sim \text{N}\left(\frac{\sum_{i=1}^K w_{ki}\phi_i}{\sum_{i=1}^K w_{ki}}, \frac{\tau^2}{\sum_{i=1}^K w_{ki}}\right) \\ \theta_k &\sim \text{N}(0, \sigma^2) \\ \tau^2, \sigma^2 &\sim \text{Inverse-Gamma}(a, b). \end{aligned} \tag{2}$$

Here $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_K)$ are independent random effects with zero mean and a constant variance, while spatial autocorrelation is modelled via random effects $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_K)$. For the latter the conditional expectation is the average of the random effects in neighbouring areas, while the conditional variance is inversely proportional to the number of neighbours. This is appropriate because if the random effects are strongly spatially autocorrelated, then the more neighbours an area has the more information there is from its neighbours about the value of its random effect, hence the uncertainty reduces. In common with the other variance parameters the default prior specification for (τ^2, σ^2) has $(a = 1, b = 0.01)$. This model contains two random effects for each data point, and as only their sum is identifiable from the data only $\psi_k = \phi_k + \theta_k$ is returned to the user.

`S.CARleroux()`

Leroux *et al.* (2000) proposed the following alternative CAR prior for modelling varying strengths of spatial autocorrelation using only a single set of random effects.

$$\begin{aligned} \psi_k &= \phi_k \\ \phi_k|\boldsymbol{\phi}_{-k}, \mathbf{W}, \tau^2, \rho &\sim \text{N}\left(\frac{\rho\sum_{i=1}^{K} w_{ki}\phi_i}{\rho\sum_{i=1}^{K} w_{ki} + 1 - \rho}, \frac{\tau^2}{\rho\sum_{i=1}^{K} w_{ki} + 1 - \rho}\right) \\ \tau^2 &\sim \text{Inverse-Gamma}(a, b) \\ \rho &\sim \text{Uniform}(0, 1). \end{aligned} \tag{3}$$

Here ρ is a spatial dependence parameter taking values in the unit interval, and can be fixed (using the argument `rho`) if required. specifically, $\rho = 1$ corresponds to the intrinsic CAR model (defined for $\boldsymbol{\phi}$ in the BYM model above), while $\rho = 0$ corresponds to independence ($\phi_k \sim \text{N}(0, \tau^2)$).

Locally smooth CAR models

The CAR priors described above enforce a single global level of spatial smoothing for the set of random effects, which for model (3) is controlled by ρ. This is illustrated by the partial autocorrelation structure implied by that model, which for (ϕ_k, ϕ_j) is given by

$$\text{COR}(\phi_k, \phi_j|\boldsymbol{\phi}_{-kj}, \mathbf{W}, \rho) = \frac{\rho w_{kj}}{\sqrt{(\rho\sum_{i=1}^{K} w_{ki} + 1 - \rho)(\rho\sum_{i=1}^{K} w_{ji} + 1 - \rho)}}. \tag{4}$$

For non-neighbouring areal units (where $w_{kj} = 0$) the random effects are conditionally independent, while for neighbouring areal units (where $w_{kj} = 1$) their partial autocorrelation is controlled by ρ. This representation of spatial smoothness is likely to be overly simplistic in practice, as the random effects surface is likely to include sub-regions of smooth evolution as well as boundaries where abrupt step changes occur. Therefore **CARBayes** can implement the localised spatial autocorrelation models proposed by Lee and Mitchell (2012) and Lee and Sarran (2015) described below.

`S.CARdissimilarity()`

Lee and Mitchell (2012) proposed a method for capturing localised spatial autocorrelation and identifying boundaries in the random effects surface. The underlying idea is to model the elements of $\mathbf{W}$ corresponding to geographically adjacent areal units as random quantities, rather than assuming they are fixed at one. Conversely, if areal units $(\mathcal{S}_k, \mathcal{S}_j)$ are not adjacent as specified by $\mathbf{W}$, then w_{kj} is fixed at zero. From (4), it is straightforward to see that if w_{kj} is estimated as one then (ϕ_k, ϕ_j) are spatially autocorrelated and are smoothed over in the modelling process, whereas if w_{kj} is estimated as zero then no smoothing is imparted between (ϕ_k, ϕ_j) as they are modelled as conditionally independent. In this case a boundary is said to exist in the random effects surface between areal units $(\mathcal{S}_k, \mathcal{S}_j)$. We note that for this model $\mathbf{W}$ must be binary.

The model is based on (3) with ρ fixed at 0.99, which ensures that the random effects exhibit strong spatial smoothing globally, which can be altered locally by estimating $\{w_{kj}|k \sim j\}$. They model each w_{kj} as a function of the dissimilarity between areal units $(\mathcal{S}_k, \mathcal{S}_j)$, because large differences in the response are likely to occur where neighbouring populations are very different. This dissimilarity is captured by q non-negative dissimilarity metrics

$\mathbf{z}_{kj} = (z_{kj1}, \ldots, z_{kjq})$, which could include social or physical factors, such as the absolute difference in smoking rates, or the proportion of the shared border that is blocked by a physical barrier (such as a river or railway line) and cannot be crossed. Using these measures of dissimilarity two distinct models are proposed for $\{w_{kj}|k \sim j\}$.

Binary model

$$\begin{aligned} w_{kj}(\boldsymbol{\alpha}) &= \begin{cases} 1 & \text{if } \exp(-\sum_{i=1}^{q} z_{kji}\alpha_i) \geq 0.5 \text{ and } k \sim j \\ 0 & \text{otherwise} \end{cases} \\ \alpha_i &\sim \text{Uniform}(0, M_i) \qquad \text{for } i = 1, \ldots, q. \end{aligned} \tag{5}$$

Non-binary model

$$\begin{aligned} w_{kj}(\boldsymbol{\alpha}) &= \exp\left(-\sum_{i=1}^{q} z_{kji}\alpha_i\right) \\ \alpha_i &\sim \text{Uniform}(0, 50) \qquad \text{for } i = 1, \ldots, q. \end{aligned} \tag{6}$$

The q regression parameters $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_q)$ determine the effects of the dissimilarity metrics on $\{w_{kj}|k \sim j\}$, and for the binary model if $\alpha_i < -\ln(0.5)/\max\{z_{kji}\}$, then the ith dissimilarity metric has not solely identified any boundaries because $\exp(-\alpha_i z_{kji}) > 0.5$ for all $k \sim j$. The upper limits M_i for the priors for α_i in the binary model depend on the distribution of z_{kji}, and are chosen to be weakly informative and fixed in the software. Users can choose between (5) and (6) by the logical argument `W.binary`, where `TRUE` corresponds to (5), while `FALSE` corresponds to (6).

`S.CARlocalised()`

An alternative to the above is to augment the set of spatially smooth random effects with a piecewise constant intercept model, thus allowing large jumps in the mean surface between adjacent areal units with different intercepts. Lee and Sarran (2015) proposed a model that partitions the K areal units into a maximum of G groups, each with their own intercept term $(\lambda_1, \ldots, \lambda_G)$. The model is given by

$$\begin{aligned} \psi_k &= \phi_k + \lambda_{Z_k} \\ \phi_k|\boldsymbol{\phi}_{-k}, \mathbf{W}, \tau^2 &\sim \text{N}\left(\frac{\sum_{i=1}^{K} w_{ki}\phi_i}{\sum_{i=1}^{K} w_{ki}}, \frac{\tau^2}{\sum_{i=1}^{K} w_{ki}}\right) \\ \tau^2 &\sim \text{Inverse-Gamma}(a, b) \\ \lambda_i &\sim \text{Uniform}(\lambda_{i-1}, \lambda_{i+1}) \qquad \text{for } i = 1, \ldots, G \\ f(Z_k) &= \frac{\exp(-\delta(Z_k - G^*)^2)}{\sum_{r=1}^{G} \exp(-\delta(r - G^*)^2)} \\ \delta &\sim \text{Uniform}(1, M). \end{aligned} \tag{7}$$

The group means $(\lambda_1, \ldots, \lambda_G)$ are ordered so that $\lambda_1 < \lambda_2 < \ldots < \lambda_G$, which prevents the label switching problem common in mixture models, and for completeness $\lambda_0 = -\infty$ and $\lambda_{G+1} = \infty$. Area k is assigned to one of the G intercepts by $Z_k \in \{1, \ldots, G\}$, and G is the maximum number of different intercept terms. Here we penalise Z_k towards the middle

intercept value, so that the extreme intercept classes (e.g. 1 or G) may be empty. This is achieved by the penalty term $\delta(Z_k - G^*)^2$ in the prior for Z_k, where $G^* = (G+1)/2$ if G is odd and $G^* = G/2$ if G is even, and is the middle group. A weakly informative uniform prior is specified for the penalty parameter $\delta \sim \text{Uniform}(1, M)$ (by default $M = 10$), so that the data play the dominant role in estimating its value. Note, a Gaussian likelihood is not allowed with this model because of a lack of identifiability among the parameters, and missing values are not allowed in the response for the same reasons.

2.2. Multivariate spatial data models

The study region $\mathcal{S}$ is again partitioned into K non-overlapping areal units $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_K\}$, and each unit contains J response variables $\mathbf{Y}_k = (Y_{k1}, \ldots, Y_{kJ})$ and J corresponding offsets $\mathbf{O}_k = (O_{k1}, \ldots, O_{kJ})$. The model therefore has to represent both spatial autocorrelation and between variable correlation, and the general multivariate spatial mixed model is given by

$$\begin{aligned} Y_{kj}|\mu_{kj} &\sim f(y_{kj}|\mu_{kj}, \nu_j^2) \quad \text{for } k = 1, \ldots, K, \quad j = 1, \ldots, J \\ g(\mu_{kj}) &= \mathbf{x}_k^\top \boldsymbol{\beta}_j + O_{kj} + \psi_{kj} \\ \boldsymbol{\beta}_j &\sim \text{N}(\boldsymbol{\mu}_\beta, \boldsymbol{\Sigma}_\beta) \\ \nu^2 &\sim \text{Inverse-Gamma}(a, b). \end{aligned} \tag{8}$$

In common with the univariate models $\mathbf{x}_k^\top$ is a vector of p covariates, and the same covariates are used for each of the J response variables. The regression coefficients $\boldsymbol{\beta}_j$ vary by response variable j, and Gaussian priors are assumed for the regression parameters $\boldsymbol{\beta}_j$ as before. The following data likelihood models are allowed:

- **Binomial -** $Y_{kj} \sim \text{Binomial}(n_{kj}, \theta_{kj})$ and $\ln(\theta_{kj}/(1-\theta_{kj})) = \mathbf{x}_k^\top \boldsymbol{\beta}_j + O_{kj} + \psi_{kj}$.
- **Gaussian -** $Y_{kj} \sim \text{N}(\mu_{kj}, \nu_j^2)$ and $\mu_{kj} = \mathbf{x}_k^\top \boldsymbol{\beta}_j + O_{kj} + \psi_{kj}$.
- **Multinomial -** For this model the response variables are the $J > 2$ multinomial categories of a single variable, where as for the other data likelihood models the J variables are distinct from each other. The data likelihood model is:

$$\begin{aligned} \mathbf{Y}_k = (Y_{k1}, \ldots, Y_{kJ}) &\sim \text{Multinomial}(n_k, \theta_{k1}, \ldots, \theta_{kJ}) \\ \ln(\theta_{kj}/\theta_{k1}) &= \mathbf{x}_k^\top \boldsymbol{\beta}_j + O_{kj} + \psi_{kj} \quad \text{for } j = 2, \ldots, J, \end{aligned}$$

  where $n_k = \sum_{j=1}^J Y_{kj}$. The above holds for categories $j = 2, \ldots, J$, and thus category $j = 1$ is a baseline and has no regression parameters or random effects or offset terms (they are all zero). Here θ_{kj} is the probability of a single outcome in area k being in category j, and hence $\sum_{j=1}^J \theta_{kj} = 1$.
- **Poisson -** $Y_{kj} \sim \text{Poisson}(\mu_{kj})$ and $\ln(\mu_{kj}) = \mathbf{x}_k^\top \boldsymbol{\beta}_j + O_{kj} + \psi_{kj}$.

When fitting these models the response variable and offset should be $K \times J$ matrices, while each covariate should be a $K \times 1$ vector. As the multinomial model models the first category as a baseline there will be $J-1$ different regression parameter sets and random effect surfaces, where as for the other data likelihood models there will be J regression parameter sets and random effect surfaces. The set of random effects are denoted by $\boldsymbol{\psi} = (\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_K)$, where $\boldsymbol{\psi}_k = (\psi_{k1}, \ldots, \psi_{kJ})$ are the set of J values ($J-1$ for the multinomial model as $\psi_{k1} = 0$) for area k. The random effects need to model both spatial autocorrelation and between variable correlation, and this is achieved using a multivariate conditional autoregressive (MCAR) model, for details see Gelfand and Vounatsou (2003). **CARBayes** can fit the following multivariate data models.

`S.glm()`
The `S.glm()` function discussed earlier can also be applied to multinomial data, where in the above equation $\psi_{kj} = 0$ for all (k, j).

`MVS.CARleroux()`
This model can be implemented with binomial, Gaussian, multinomial and Poisson data likelihoods. The random effects $\boldsymbol{\psi}$ are equal to a single component $\boldsymbol{\phi}$ (e.g. $\boldsymbol{\psi} = \boldsymbol{\phi}$), and are modelled using the approach outlined in Kavanagh *et al.* (2016) given by:

$$\boldsymbol{\phi} \sim \mathrm{N}\left(\mathbf{0}, \left[\mathbf{Q}(\mathbf{W}, \rho) \otimes \boldsymbol{\Sigma}^{-1}\right]^{-1}\right). \tag{9}$$

Here $\mathbf{Q}(\mathbf{W}, \rho) = \rho[\mathrm{diag}(\mathbf{W1}) - \mathbf{W}] + (1-\rho)\mathbf{I}$ is the precision matrix for the joint distribution corresponding to the CAR prior proposed by Leroux *et al.* (2000) and described above, while $\boldsymbol{\Sigma}_{J \times J}$ is a cross variable covariance matrix. In common with the univariate models, the correlation structure imposed by (9) is more easily seen by its full conditional form, that is:

$$\begin{aligned}
\boldsymbol{\phi}_k | \boldsymbol{\phi}_{-k}, \mathbf{W}, \boldsymbol{\Sigma}, \rho &\sim \mathrm{N}\left(\frac{\rho \sum_{i=1}^{K} w_{ki} \boldsymbol{\phi}_i}{\rho \sum_{i=1}^{K} w_{ki} + 1 - \rho}, \frac{\boldsymbol{\Sigma}}{\rho \sum_{i=1}^{K} w_{ki} + 1 - \rho}\right) \\
\boldsymbol{\Sigma} &\sim \text{Inverse-Wishart}(df, \boldsymbol{\Omega}) \\
\rho &\sim \text{Uniform}(0, 1),
\end{aligned}$$

where $\boldsymbol{\phi}_{-k}$ denotes the vector of random effects except those relating to the kth areal unit. We specify the marginally weakly-informative prior for $\boldsymbol{\Sigma}$ that was designed for covariance matrices by Huang and Wand (2013), which comes from the Inverse-Wishart family with degrees of freedom $df = \nu + J - 1$ and scale $\boldsymbol{\Omega} = 2\nu \mathrm{diag}(1/A_1, \ldots, 1/A_J)$, where J is the number of outcome variables being modelled. Here, $\mathrm{diag}(1/A_1, \ldots, 1/A_J)$ denotes a diagonal matrix with diagonal entries $(A_1, \ldots, A_J)$. Thus in the prior specification the user can specify the degrees of freedom ν 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 jth vector of random effects $\sqrt{\boldsymbol{\Sigma}_{jj}}$ is modelled by a zero centred half-t(ν, A_j) distribution with degrees of freedom ν 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 $\boldsymbol{\Sigma}_{ij}/\sqrt{\boldsymbol{\Sigma}_{ii}\boldsymbol{\Sigma}_{jj}}$, 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.

Finally, in common with the univariate model `S.CARleroux()`, the spatial autocorrelation parameter ρ can be fixed to any value in the unit interval using the argument `rho`.

2.3. Two-level spatial data models

The study region $\mathcal{S}$ is again partitioned into K non-overlapping areal units $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_K\}$, and data are available on m_k individuals within area k. Thus for areal unit $\mathcal{S}_k$ there are m_k different response variables being modelled, leading to both spatial variation and individual-level variation. The general likelihood model allowed for these data is given by

$$\begin{aligned} Y_{kj}|\mu_{kj} &\sim f(y_{kj}|\mu_{kj}, \nu^2) \quad \text{for } k = 1, \ldots, K, \quad j = 1, \ldots, m_k, \\ g(\mu_{kj}) &= \mathbf{x}_{kj}^\top \boldsymbol{\beta} + O_{kj} + \psi_k, \\ \boldsymbol{\beta} &\sim \text{N}(\boldsymbol{\mu}_\beta, \boldsymbol{\Sigma}_\beta) \\ \nu^2 &\sim \text{Inverse-Gamma}(a, b). \end{aligned} \tag{10}$$

In common with the univariate models $(\mathbf{x}_{kj}^\top, O_{kj})$ are respectively a vector of p covariates and an offset for individual j within area k. For this model the response and each covariate vector is of length $m = \sum_{k=1}^K m_k$. Gaussian priors are again assumed for the regression parameters $\boldsymbol{\beta}$. Binomial, Gaussian and Poisson data likelihood models are allowed, that is:

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

CARBayes can only fit the following model for ψ_k.

`S.CARmultilevel()`
This model can be implemented with binomial, Gaussian and Poisson data likelihoods and has the following random effects structure:

$$\begin{aligned} \psi_{kj} &= \phi_k, \\ \phi_k|\boldsymbol{\phi}_{-k} &\sim \text{N}\left(\frac{\rho \sum_{j=1}^K w_{kj}\phi_j}{\rho \sum_{j=1}^K w_{kj} + 1 - \rho}, \frac{\tau^2}{\rho \sum_{j=1}^K w_{kj} + 1 - \rho}\right), \\ \tau^2 &\sim \text{Inverse-Gamma}(a, b). \\ \rho &\sim \text{Uniform}(0, 1). \end{aligned} \tag{11}$$

The spatial variation is modelled by $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_K)$, which is common to all individuals within each area and is modelled by the CAR prior proposed by Leroux *et al.* (2000). Again

ρ can be fixed to any value in the unit interval using the argument `rho`. The ordering of the response and covariate data vectors are not constrained to have all individuals in area 1 followed by all individuals in area 2, etc. Instead, the `S.CARmultilevel()` function requires the `ind.area` argument to be specified, which is a vector of length m. Each element in that vector must be an integer between 1 and K (where K is the number of areas), and denotes which area an individual belongs to as ordered by the **W** matrix. For example, if the rth element of `ind.area` is 5, then the rth element in each response and covariate data vector refers to an individual in area 5, that is the area represented by the 5th row of the neighbourhood matrix **W**.

2.4. Inference

All models in this package are fitted in a Bayesian setting using MCMC simulation, and the new functionality in version 6.0 is that you can now generate multiple MCMC chains in one model run. 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 MCMC samples are generated via a combination of Gibbs sampling (when the appropriate full conditional distributions are proportional to standard distributions) and Metropolis-Hastings steps. The Metropolis-Hastings steps for the regression parameters use simple random walk Metropolis steps by setting `MALA=FALSE` in the function call, or the Metropolis adjusted Langevin algorithm (MALA, Roberts and Rosenthal 1998) if you set `MALA=TRUE`, the latter being the default. 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 matrix **W** is utilised via its triplet form when updating the random effects within the algorithms, which increases the computational efficiency of the software. Additionally, matrix identities and Kronecker product forms are used to speed up the computation where possible. Missing values are allowed in the response variable **Y** for most models (not the `S.CARlocalised()` model), and are treated as additional parameters to be updated in the MCMC algorithm using a data augmentation approach (Tanner and Wong 1987).

3. Loading and using the software

3.1. Loading the software

CARBayes is an add-on package to the statistical software R, and is freely available to download from the Comprehensive R Archive Network (CRAN, http://cran.r-project.org/) for Windows, Linux and Apple platforms. The package requires R ($\geq$ 3.5.0) and depends on packages **MASS** (Venables and Ripley 2002), and **Rcpp** ($\geq$ 0.11.5). Additionally, it imports functionality from the following other packages: **CARBayesdata** (Lee 2022), **coda** (Plummer *et al.* 2006), **mapview** (Appelhans *et al.* 2022), **matrixcalc** (Novomestky 2012), **MCMCpack** (Martin *et al.* 2011), **parallel**, **RColorBrewer** (Neuwirth 2022), **sf** (Pebesma 2018), **spam** (Furrer and Sain 2010), **spdep** (Bivand *et al.* 2013), **stats**, **truncnorm** (Trautmann *et al.* 2014) and **utils**. Once installed it can be loaded using the command `library(CARBayes)`. Note, certain functionality from the packages listed in the previous paragraph are automatically

loaded upon loading **CARBayes**, but only for use within the package. However, a complete spatial analysis will typically also include the creation of the neighbourhood matrix **W** from a shapefile, the production of spatial maps of the fitted values and residuals, and tests for the presence of spatial autocorrelation, which are achieved using the other packages.

3.2. Using the software

The software can fit seven models: `S.glm()`, `S.CARbym()`, `S.CARleroux()`, `S.CARdissimilarity()` and `S.CARlocalised()` for univariate spatial data, `MVS.CARleroux()` for multivariate spatial data, and `S.CARmultilevel()` for two-level data relating to individuals within areas. Full details of the arguments required for each model are given in the help files. However, the main common arguments that are required for a baseline analysis (for example using default priors) are as follows.

- `formula` - A formula for the covariate part of the model using the same syntax as the `lm()` function. Offsets can be included here using the `offset()` function.
- `family` - The data likelihood model which must be one of `"binomial"`, `"gaussian"`, `"multinomial"`, `"poisson"` or `"zip"`.
- `trials` - This is only needed if `family="binomial"` or `family="multinomial"`, and is a vector the same length as the response containing the total number of trials for each area.
- `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).
- `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 **CARBayes**, the software provides the following summary extractor functions:

- `fitted()` - returns the fitted values based on the posterior mean.
- `logLik()` - returns the estimated loglikelihood based on the posterior mean.
- `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 *CARBayesprogress.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 of 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 acceptance rates of the Markov chains, the number of samples generated after burnin and thinning, the effective number of independent samples using the `effectiveSize()` function from the **coda** package (`n.effective`), and a convergence diagnostic for the Markov chains also from the **coda** package. If more than one chain is generated by the model (i.e. if `n.chains` is greater than 1) then this diagnostic is the Upper C.I. of the potential scale reduction factor (PSRF, Gelman *et al.* 2003), and a a value less than 1.1 is indicative of convergence. If only 1 Markov chain is generated by the model (i.e. if `n.chains` equals 1) then this diagnostic is the Z-score statistic proposed by Geweke (1992), and values within the range (-1.96, 1.96) are indicative of convergence.

- `samples` - A list containing the MCMC samples generated from the model, where each element in the list is a matrix. The names of these matrix objects correspond to the parameters defined in Section 2 of this vignette, and each column of a matrix contains the set of samples for a single parameter. This list includes samples from the posterior distribution of the fitted values for each data point (`fitted`). Additionally, if the response variable $\mathbf{Y}$ contains missing values, then samples from its posterior predictive distribution obtained via data augmentation are available (`Y`).

- `fitted.values` - The fitted values based on the posterior mean from the model. For the univariate data models this is a vector, while for the multivariate data models this is a matrix.

- `residuals` - For the univariate data models this is a matrix with 2 columns, where each column is a type of residual and each row relates to a single data point. The types are *response* (raw) and *pearson*. For the multivariate data models this is a list with 2 $K \times J$ matrix elements, where each matrix element is a type of residual (*response* or *pearson*).

- `modelfit` - Model fit criteria including the Deviance Information Criterion (DIC, Spiegelhalter *et al.* 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 contains missing data, the DIC is computed based on only the observed data (see Celeux *et al.* 2006).

- `accept` The acceptance probabilities for the parameters after burnin.
- `localised.structure` - This element is `NULL` except for the models `S.CARdissimilarity()` and `S.CARlocalised()`. For `S.CARdissimilarity` it is a list containing two matrices, `W.posterior` and `W.border.prob`. `W.posterior` contains posterior medians for each element w_{kj} of the $K \times K$ neighbourhood matrix $\mathbf{W}$, while `W.border.prob` contains posterior probabilities that each w_{kj} element equals zero, which corresponds to the posterior probability of a boundary in the random effects surface. The latter is only present if `W.binary=TRUE`, otherwise it is missing (`NA`). In all cases elements of $\mathbf{W}$ that correspond to non-neighbouring areas as determined by the original $\mathbf{W}$ matrix have `NA` values. For `S.CARlocalised()` this element is a vector of length K, and gives the posterior median class (Z_k value) that each data point is assigned to.
- `formula` - The formula (as a text string) for the response, covariate and offset part of the model.
- `model`- A text string describing the model that has been fitted.
- `mcmc.info` - A vector giving details of the numbers of MCMC samples generated.
- `X` - The design matrix of covariates inherited from the `formula` argument.

The remainder of this vignette illustrates the **CARBayes** software via 2 worked examples.

4. Example 1 - property prices in Greater Glasgow

The **CARBayes** software is illustrated by modelling the spatial pattern in average property prices across Greater Glasgow, Scotland, in 2008. This is an ecological regression analysis, whose aim is to identify the factors that affect property prices and quantify their effects.

4.1. Data and exploratory analysis

The data come from the Scottish Statistics database (`http://statistics.gov.scot`), but are also included in the **CARBayesdata** R package. The study region is the Greater Glasgow and Clyde health board (GGHB), which is split into 271 Intermediate zones (IZ). These IZs are small areas that have a median population of 4,239. These data can be loaded into R using the code below:

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

Then the first 6 rows of each data set can be viewed using the following code.

```
R> head(pricedata)
```

```
         IZ   price crime rooms sales driveshop     type
1 S02000260 112.250   390     3    68       1.2     flat
2 S02000261 156.875   116     5    26       2.0     semi
3 S02000262 178.111   196     5    34       1.7     semi
4 S02000263 249.725   146     5    80       1.5 detached
5 S02000264 174.500   288     4    60       0.8     semi
6 S02000265 163.521   342     4    24       2.5     semi
```

```
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
```

The `GGHB.IZ` object is of type `sf` and contains the spatial information for the GGHB, which is used to map the data, construct the neighbourhood matrix $\mathbf{W}$, and conduct a test for spatial autocorrelation. The `pricedata` object is a `data.frame` containing the property price data for 270 of the 271 IZs in GGHB, because one area had outlying values and was hence removed. The variables in `pricedata` are highlighted in the output above, and the `IZ` column contains unique identifiers for each IZ. The data are summarised in Table 2, which displays the percentiles of their distribution (with the exception of the categorical variable `type`).

The response variable in this study is the median price (in thousands, `price`) of all properties sold in 2008 in each IZ. The table shows large variation in this variable, with average prices ranging between £50,000 and £372,800 across the study region. The first covariate in this study is the crime rate (`crime`) in each IZ, because areas with higher crime rates are likely to be less desirable to live in. Crime rate is measured as the total number of recorded crimes in each IZ per 10,000 people that live there, and the values range between 85 and 1994. Other covariates included in this study are the median number of rooms in a property (`rooms`), the number of properties that sold in a year (`sales`), and the average time taken to drive to the nearest shopping centre (`driveshop`). The latter is a proxy measure of

Variable	**Percentiles**				
	0%	25%	50%	75%	100%
Property price (in thousands)	50.0	95.0	121.8	159.2	372.8
Crime rate (per 10,000 population)	85.0	303.2	517.0	728.0	1994.0
Number of rooms (median)	3	3	4	4	6
Property sales	4	46	58	85	266
Drive time to a shop (minutes)	0.3	0.8	1.2	1.9	8.5

Table 2: Summary of the distribution of the data.

access to services which may affect property prices. Finally, a categorical variable measuring the most prevalent property type in each area is available (`type`), with levels; 'flat' (68% of areas), 'terraced' (7%), 'semi-detached' (13%) and 'detached' (12%).

Property prices are positive and skewed to the right, and an initial linear regression model including all the covariates showed the residuals from this model were non-normal and skewed to the right. Therefore we model property price on the natural log scale, and the log transformed variable can be added to the data set using the code below.

```
R> library(dplyr)
R> pricedata <- pricedata %>% mutate(logprice = log(pricedata$price))
R> head(pricedata)

         IZ   price crime rooms sales driveshop     type logprice
1 S02000260 112.250   390     3    68       1.2     flat 4.720729
2 S02000261 156.875   116     5    26       2.0     semi 5.055449
3 S02000262 178.111   196     5    34       1.7     semi 5.182407
4 S02000263 249.725   146     5    80       1.5 detached 5.520360
5 S02000264 174.500   288     4    60       0.8     semi 5.161925
6 S02000265 163.521   342     4    24       2.5     semi 5.096941
```

Then the relationships between these variables can be visualised using the **GGally** package as follows, and the result is displayed in Figure 1.

```
R> library(GGally)
R> ggpairs(data = pricedata, columns = c(8, 3:7))
```

The figure shows a negative relationship between log property price and crime rate, while positive relationships exist between log property price and both the number of rooms and the sales rate. Additionally, property type has a strong relationship with log price, with areas having the most detached properties having the highest prices. The next step is to produce a map of property prices across the Greater Glasgow and Clyde region, but to do this the data must be merged with the `sf` object `GGHB.IB`. This merging can be done as follows:

```
R> pricedata.sf <- merge(x=GGHB.IZ, y=pricedata, by="IZ", all.x=FALSE)
```

Then a spatial map of the `price` variable can be overlaid on a map of the region using the functionality of the **mapview** package. However, first the `pricedata.sf` object needs to have

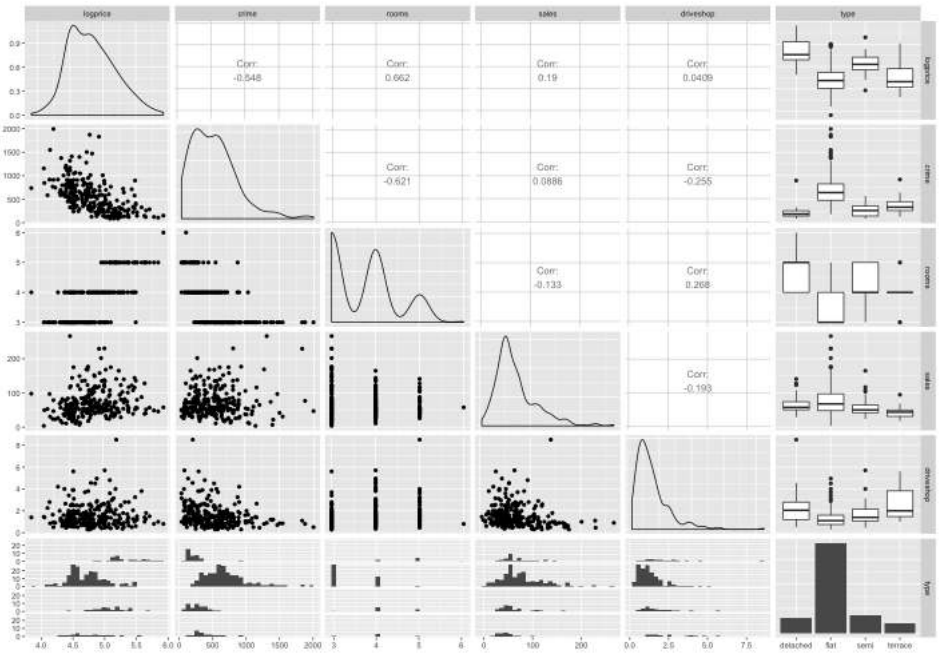

Figure 1: Scatterplot showing the relationships in the data.

its coordinate reference system changed to longitude and latitude as this is what the **mapview** package requires, which can be done using the following R code.

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

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

```
R> library(mapview)
R> library(RColorBrewer)
R> map1 <- mapview(pricedata.sf, zcol = "price", col.regions=brewer.pal(9, "YlOrRd"),
+             alpha.regions=0.6, layer.name="Price", lwd=0.5, col="grey90",
+             homebutton=FALSE)
R> removeMapJunk(map1, junk = c("zoomControl", "layersControl"))
```

The map is shown in Figure 2 and suggests that Glasgow has a number of property submarkets, whose prices are not related to those in neighbouring areas. An example of this is the two groups of higher priced regions north of the river Clyde, which are the highly sought after Westerton / Bearsden (northerly cluster) and Dowanhill / Hyndland (central cluster) districts.

4.2. Non-spatial modelling

The natural log of the median property price is treated as the response and assumed to be Gaussian, and an initial covariate only model is built in a frequentist framework using linear models. A model with all the covariates is fitted to the data using the model below.

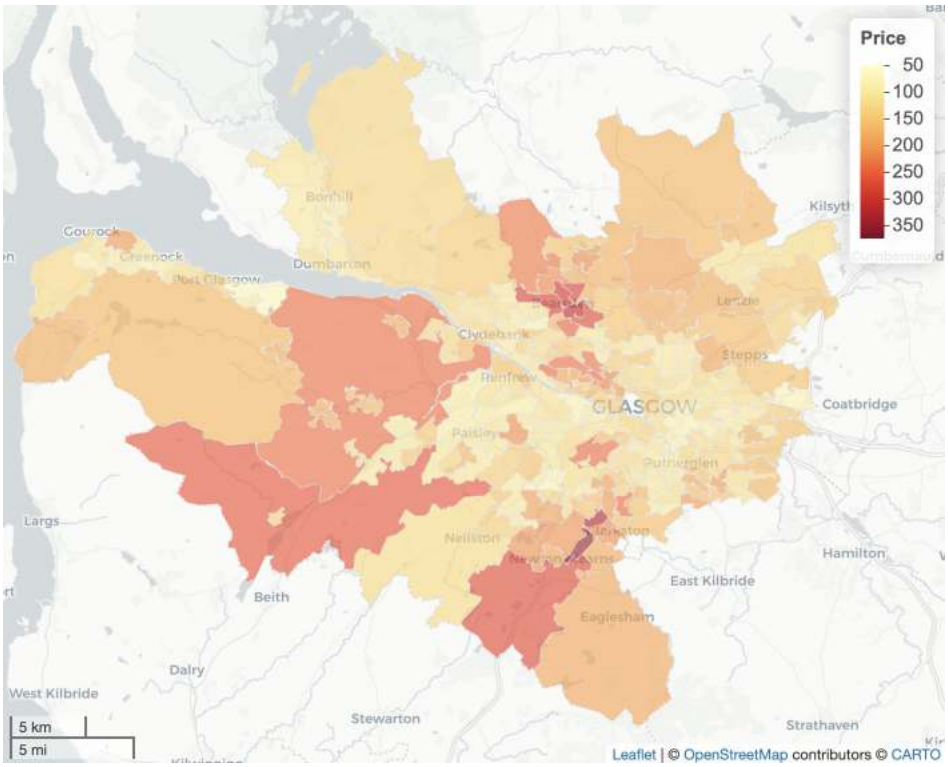

Figure 2: Map showing the average property price in each IG (in thousands).

```
R> form <- logprice~crime+rooms+sales+factor(type) + driveshop
R> model <- lm(formula=form, data=pricedata.sf)
R> summary(model)


Call:
lm(formula = form, data = pricedata.sf)

Residuals:
    Min      1Q  Median      3Q     Max
-0.9717 -0.1510 -0.0012  0.1672  0.8346

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)          4.342e+00  1.540e-01  28.203  < 2e-16 ***
crime               -2.248e-04  5.541e-05  -4.057 6.55e-05 ***
rooms                2.119e-01  2.890e-02   7.334 2.79e-12 ***
sales                2.264e-03  3.544e-04   6.388 7.61e-10 ***
factor(type)flat    -2.948e-01  5.988e-02  -4.923 1.51e-06 ***
factor(type)semi    -1.768e-01  5.747e-02  -3.077 0.002312 **
factor(type)terrace -3.251e-01  6.921e-02  -4.697 4.26e-06 ***
driveshop           -4.848e-02  1.452e-02  -3.340 0.000961 ***
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2273 on 262 degrees of freedom
Multiple R-squared:  0.6073,        Adjusted R-squared:  0.5968
F-statistic: 57.88 on 7 and 262 DF,  p-value: < 2.2e-16
```

From the model output above all of the covariates are significantly related to the response at the 5% level, suggesting they all play an important role in explaining the spatial pattern in median property price. 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 to assess its significance. 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 using the code below. Lines 2 and 3 turn `pricedata.sf` into a neighbourhood (`nb`) object and then into a `listw` object, which is the required form of the binary spatial adjacency information (based on border sharing) used by the `moran.mc()` function.

```
R> library(spdep)
R> W.nb <- poly2nb(pricedata.sf, row.names = pricedata.sf$IZ)
R> W.list <- nb2listw(W.nb, style="B")
R> moran.mc(x=residuals(model), listw=W.list, nsim=1000)

        Monte-Carlo simulation of Moran I

data:  residuals(model)
weights: W.list
number of simulations + 1: 1001

statistic = 0.30141, observed rank = 1001, p-value = 0.000999
alternative hypothesis: greater
```

The Moran's I test has a p-value much less than 0.05, which suggests that the residuals contain substantial positive spatial autocorrelation.

4.3. Spatial modelling with CARBayes

The residual spatial autocorrelation observed above can be accounted for by adding a set of random effects to the model, and we apply the `S.CARleroux()` model (equations (1) and (3)) to the data to account for this. However, first we need to create the neighbourhood matrix $\mathbf{W}$ as shown below.

```
R> W <- nb2mat(W.nb, style="B")
```

Then the model can be run with 3 parallel Markov chains on 3 processors of the same computer as follows, which is much computationally quicker than running the 3 chains one after the other.

```
R> library(CARBayes)
R> chain <- S.CARleroux(formula=form, data=pricedata.sf, family="gaussian", W=W,
+       burnin=100000, n.sample=300000, thin=100, n.chains=3, n.cores=3)
```

In the above model run the covariate and offset component defined by `formula` is the same as for the simple linear model fitted earlier, and the neighbourhood matrix `W` is binary and defined by whether or not two areas share a common border. Inference for this model is based on 3 parallel Markov chains, each of which has been run for 300,000 samples, the first 100,000 of which have been removed as the burn-in period. The remaining 200,000 samples are thinned by 100 to reduce their temporal autocorrelation, resulting in 6,000 samples for inference across the 3 Markov chains. A summary of the model output can be obtained using the `print()` function as shown below.

```
R> print(chain1)

#################
#### Model fitted
#################
Likelihood model - Gaussian (identity link function)
Random effects model - Leroux CAR
Regression equation - logprice ~ crime + rooms + sales + factor(type) + driveshop

#################
#### 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 - 1e+05
Amount of thinning used - 100

############
#### Results
############
Posterior quantities and DIC

                      Mean    2.5%   97.5% n.effective PSRF (upper 95% CI)
(Intercept)         4.1337  3.8640  4.4063      6000.0                    1
crime              -0.0001 -0.0002 -0.0001      6000.0                    1
rooms               0.2336  0.1816  0.2840      6000.0                    1
sales               0.0023  0.0017  0.0029      6143.8                    1
factor(type)flat   -0.2946 -0.4066 -0.1862      5792.9                    1
factor(type)semi   -0.1714 -0.2676 -0.0721      5980.2                    1
factor(type)terrace -0.3236 -0.4475 -0.2058      6000.0                    1
driveshop           0.0036 -0.0309  0.0381      6015.6                    1
nu2                 0.0224  0.0114  0.0324      6000.0                    1
tau2                0.0536  0.0243  0.0964      5822.4                    1
rho                 0.9110  0.7272  0.9906      6000.0                    1
```

```
DIC =  -162.4       p.d =  104.7452       LMPL =  58.42
```

This model summary is presented in three parts, the first of which describes the model that has been fitted while the second describes the number of MCMC samples generated. The third part summarises the results, and includes the following information for selected parameters (not the random effects): (i) posterior mean (`Mean`); (ii) 95% credible intervals (`2.5%, 97.5%`); (iii) the effective number of independent samples (`n.effective`); and (iv) the potential scale reduction factor (PSRF) upper C.I. proposed by Gelman *et al.* (2003). Finally, the DIC and LMPL model fit criteria are displayed. In addition to producing the summary above, the model returns a list object with the following components:

```
R> summary(chain1)
```

```
                    Length Class      Mode
summary.results       77   -none-     numeric
samples                7   -none-     list
fitted.values        270   -none-     numeric
residuals              2   data.frame list
modelfit               6   -none-     numeric
accept                 5   -none-     numeric
localised.structure    0   -none-     NULL
formula                3   formula    call
model                  2   -none-     character
mcmc.info              4   -none-     numeric
X                   2160   -none-     numeric
```

The first element is the summary results table used by the `print()` function. The next element is a list containing matrices of the thinned and post burn-in MCMC samples for each set of parameters. The next two elements in the list `fitted.values` and `residuals` contain the fitted values and residuals from the model, while `modelfit` gives a selection of model fit criteria. These criteria include the Deviance Information Criterion (DIC), the log Marginal Predictive Likelihood (LMPL), the Watanabe-Akaike Information Criterion (WAIC), and the log likelihood. For further details about Bayesian modelling and model fit criteria see Gelman *et al.* (2003). The item `accept` contains the acceptance rates for the Markov chain, while `localised.structure` is `NULL` for this model and is used for compatibility with the other functions in the package. Finally, the `formula` and `model` elements are text strings describing the formula used and the model fit, `mcmc.info` details the numbers of MCMC samples generated, while `X` gives the design matrix corresponding to the `formula` object.

Before one can draw inference from the model, the convergence of the MCMC samples needs to be assessed. This should be done for all parameters in the model, but as this is impractical for the large number of random effects, a subset of the random effects is often checked. The samples from the model run are stored in the `samples` element of the model object, which is a list as shown below

```
R> summary(chain1$samples)
```

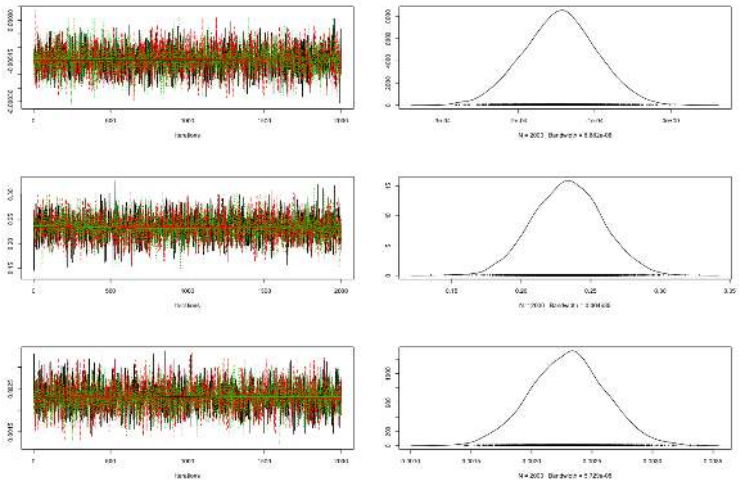

Figure 3: Convergence of the Markov chains.

```
       Length Class      Mode
beta   3      mcmc.list list
phi    3      mcmc.list list
rho    3      mcmc.list list
tau2   3      mcmc.list list
nu2    3      mcmc.list list
fitted 3      mcmc.list list
Y      3      mcmc.list list
```

Each element of this list corresponds to a different group of parameters, and is stored as either an `mcmc` object if `n.chains` equals 1 or an `mcmc.list` object if `n.chains` is greater than 1. Both these object types come from the **coda** package. Here the `Y` object is `NA` as there are no missing Y_k observations in this data set. If there had been say m missing values, then the `Y` component of the list would have contained m columns, with each one containing posterior predictive samples for one of the missing observations. Before making inference from the model you have to ensure the Markov chains appear to have converged, and as described above the potential scale reduction factor (PSRF) is given in the model summary above (`PSRF (upper 95% CI)`). As this is less than 1.1 in all cases then convergence is indicated. Another method for assessing MCMC convergence is a traceplot comparing the results from the multiple chains, which for the regression parameters (`beta`) can be produced as follows.

```
R> library(coda)
R> plot(beta.samples[ ,2:4])
```

A plot of the samples for the $2nd$ (crime) to the $4th$ (sales) regression parameters 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.

4.4. Inference

For this model inference centres around the regression parameters $\boldsymbol{\beta}$, and the samples are combined across all 3 Markov chains and summarised using the following code

```
R> summary.beta <- summary(chain$samples$beta, quantiles=c(0.025, 0.975))
```

Then from this summary output a table of posterior means and 95% credible intervals can be computed as follows:

```
R> beta.mean <- summary.beta$statistics[ ,"Mean"]
R> beta.ci <- summary.beta$quantiles
R> beta.results <- cbind(beta.mean, beta.ci)
R> rownames(beta.results) <- colnames(chain$X)
R> round(beta.results, 5)
```

```
                    beta.mean     2.5%    97.5%
(Intercept)           4.13372  3.86401  4.40626
crime                -0.00014 -0.00024 -0.00005
rooms                 0.23364  0.18157  0.28404
sales                 0.00231  0.00169  0.00294
factor(type)flat     -0.29463 -0.40660 -0.18623
factor(type)semi     -0.17144 -0.26762 -0.07206
factor(type)terrace  -0.32364 -0.44746 -0.20580
driveshop             0.00355 -0.03087  0.03815
```

The results show a significant negative effect of crime rate on property price, while increases in the number of rooms and sales rates increases property prices. Areas where the predominant property type is 'Detached' (baseline level) have higher prices than the other three types, because the latter all have significantly negative regression effects. Finally, the time taken to drive to a shop has no significant effect on property prices because the 95% credible interval contains zero.

5. Example 2 - identifying high-risk disease clusters

The third example illustrates the utility of the localised spatial autocorrelation model proposed by Lee and Mitchell (2012), which can identify boundaries that represent step changes in the (random effects) response surface between geographically adjacent areal units. The aim of this analysis is to identify boundaries in the risk surface of respiratory disease in Greater Glasgow, Scotland, in 2010, so that the spatial extent of high-risk clusters can be identified. The identification of boundaries in spatial data is affectionately known as *Wombling*, after the seminal paper by Womble (1951).

5.1. Data and exploratory analysis

The data again relate to the Greater Glasgow and Clyde health board, and are also freely available to download from Scottish Statistics (http://statistics.gov.scot). However,

the river Clyde partitions the study region into northern and southern sub-regions, and no areal units on opposite banks of the river border each other. This means that boundaries could not be identified across the river, and therefore here we only consider those areal units that are on the northern side of the study region. This leaves $K = 134$ areal units in the new smaller study region, and data on respiratory disease for this region are included in the **CARBayesdata** package and can be loaded with the command:

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

Then the first 6 rows of each data set can be viewed using the following code.

```
R> head(pricedata)
```

```
         IZ   price crime rooms sales driveshop     type logprice
1 S02000260 112.250   390     3    68       1.2     flat 4.720729
2 S02000261 156.875   116     5    26       2.0     semi 5.055449
3 S02000262 178.111   196     5    34       1.7     semi 5.182407
4 S02000263 249.725   146     5    80       1.5 detached 5.520360
5 S02000264 174.500   288     4    60       0.8     semi 5.161925
6 S02000265 163.521   342     4    24       2.5     semi 5.096941
```

```
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
```

The first column contains the unique Intermediate Zone codes (`IZ`), while the observed (`observed`) and expected (`expected`) numbers of hospital admissions in 2010 in each IZ due to respiratory disease (International Classification of Disease tenth revision codes J00-J99) are stored columns 2 and 3 respectively. The latter is computed using indirect standardisation, and is included to control for varying population sizes and demographics across the IZs. Column 4 contains a measure of the percentage of people defined to be income deprived (in receipt of means tested benefits) in each IZ (`incomedep`), which is the mechanism by which boundaries will be identified. Finally, the `SMR` column measures disease risk and is the ratio of observed / expected. These data can be merged with the spatial IZ polygons to create a `sf` object using the following code.

```
R> respiratorydata.sf <- merge(x=GGHB.IZ, y=respiratorydata, by="IZ", all.x=FALSE)
R> head(respiratorydata.sf)

Simple feature collection with 6 features and 8 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  easting northing observed  expected
1 S02000260                    Auchinairn 261624.5 669657.4      107 106.45661
2 S02000261                 Woodhill East 262927.1 670027.8       23  50.97354
3 S02000262                 Woodhill West 262142.9 670428.0       53 104.49236
4 S02000263                Westerton East 254570.5 670593.8       40  90.35747
5 S02000264 Bishopbriggs West and Cadder 261248.4 670928.0       60 140.16546
6 S02000265                Westerton West 253764.4 670982.6       25  63.93549
  incomedep       SMR                         geometry
1        22 1.0051044 MULTIPOLYGON (((260718 6698...
2         7 0.4512145 MULTIPOLYGON (((262047 6699...
3         6 0.5072141 MULTIPOLYGON (((261158.7 67...
4         5 0.4426861 MULTIPOLYGON (((253771 6709...
5         7 0.4280655 MULTIPOLYGON (((260183 6706...
6         6 0.3910191 MULTIPOLYGON (((253252 6710...
```

The next step is to map the SMR, but before this can be done the coordinate reference system of `respiratorydata.sf` needs to be changed to longitude and latitude as follows.

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

Then the map can be drawn using **mapview** via the following code, which is displayed in Figure 4.

```
R> library(mapview)
R> library(RColorBrewer)
R> map2 <- mapview(respiratorydata.sf, zcol = "SMR", col.regions=brewer.pal(9, "YlOrRd"),
```

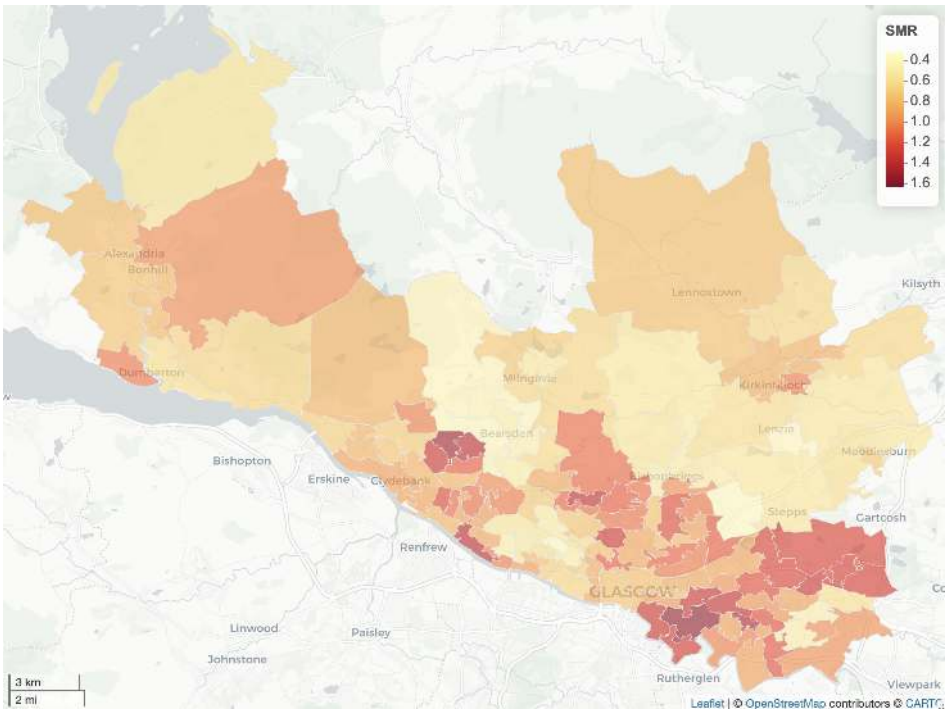

Figure 4: Map displaying the SMR for each area.

```
+              alpha.regions=0.6, layer.name="SMR", lwd=0.5, col="grey90",
+              homebutton=FALSE)
R> removeMapJunk(map2, junk = c("zoomControl", "layersControl"))
```

Values of the SMR above one relate to areas exhibiting above average risks, while values below one correspond to areas with below average risks. For example, an SMR of 1.2 corresponds to a 20% increased risk relative to the expected numbers of respiratory disease cases. The figure shows evidence of localised spatial structure, with numerous different locations where high and low risk areas border each other. This in turn suggests that boundaries are likely to be present in the risk surface, and their identification is the goal of this analysis. The method proposed by Lee and Mitchell (2012) identifies these boundaries using dissimilarity metrics, which are non-negative measures of the dissimilarity between all pairs of adjacent areas. In this example we use the absolute difference in the percentage of people in each IZ who are defined to be income deprived (`incomedep`), because it is well known that socio-economic deprivation plays a large role in determining people's health. However, before fitting the model the spatial neighbourhood matrix $\mathbf{W}$ based on sharing a common border is computed using the following code.

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

5.2. Spatial modelling with CARBayes

Let the observed and expected numbers of hospital admissions for all K IZs be denoted by

$\mathbf{Y} = (Y_1, \ldots, Y_K)$ and $\mathbf{e} = (e_1, \ldots, e_K)$ respectively. Then as the observed numbers of hospital admissions are counts, the following Poisson log-linear model is suitable for these data

$$\begin{aligned} Y_k &\sim \text{Poisson}(e_k\theta_k) \\ \ln(\theta_k) &= \beta_0 + \phi_k, \end{aligned}$$

where θ_k represents disease risk in areal unit $\mathcal{S}_k$. For a general review of disease mapping see Wakefield (2007). We note that when fitting this model in **CARBayes** the offset is specified on the linear predictor scale rather than the expected value scale, so in this analysis the offset is log($\mathbf{e}$) rather than $\mathbf{e}$. The dissimilarity metric used here is the absolute difference in the level of income deprivation between each pair of areal units, which can be created from the vector of area level income deprivation scores using the following code.

```
R> income <- respiratorydata.sf$incomedep
R> Z.incomedep <- as.matrix(dist(income, diag=TRUE, upper=TRUE))
```

The first line pulls out the income deprivation covariate while the second line computes the $K \times K$ matrix of absolute differences in income deprivation levels between each pair of areal units, that is `Z.incomedep`$_{kj}$ = |`income`$_k$ - `income`$_j$|. The function to implement the localised CAR model is called `S.CARdissimilarity()`, and it takes the same arguments as the other CAR models except that it additionally requires the dissimilarity metrics. These are required in the form of a list of $K \times K$ matrices, and for this example we only have a single dissimilarity metric. Additionally, we add the argument `W.binary=TRUE`, so that the estimated elements in $\mathbf{W}$ are ones or zeros (corresponding to (5)), the latter corresponding to boundaries. The model is run using the following code.

```
R> formula <- observed ~ offset(log(expected))
R> chain1 <- S.CARdissimilarity(formula=formula, data=respiratorydata.sf,
+                family="poisson", W=W, Z=list(Z.incomedep=Z.incomedep),
+                W.binary=TRUE, burnin=100000, n.sample=300000, thin=20)
```

Here we only use 1 Markov chain (the default value of `n.chains` is 1 so this argument can be omitted) to illustrate the slightly altered form of the summary table in this case. Here convergence of the Markov chain can be assessed using traceplots and the Geweke diagnostic outlined in the previous section, which are not shown for brevity. The above model bases inference on 10,000 post burn-in and thinned MCMC samples. A summary of the model can be obtained from the `print()` function as follows.

```
R> print(chain1)

#################
#### Model fitted
#################
Likelihood model - Poisson (log link function)
Random effects model - Binary dissimilarity CAR
```

```
Dissimilarity metrics -  Z.incomedep
Regression equation - observed ~ offset(log(expected))

#################
#### MCMC details
#################
Total number of post burnin and thinned MCMC samples generated - 10000
Number of MCMC chains used - 1
Length of the burnin period used for each chain - 1e+05
Amount of thinning used - 20

############
#### Results
############
Posterior quantities and DIC

               Mean    2.5%   97.5% n.effective Geweke.diag alpha.min
(Intercept) -0.2195 -0.2419 -0.1969     10000.0         1.1        NA
tau2         0.1373  0.0964  0.1897      9470.5         1.6        NA
Z.incomedep  0.0498  0.0465  0.0513      9401.4        -2.3    0.0139

DIC =  1058.256       p.d =  99.07107       LMPL =  -570.44

The number of stepchanges identified in the random effect surface
     no stepchange stepchange
[1,]           261         99
```

The model represents the log risk surface $\{\ln(\theta_k)\}$ with an intercept term and the random effects, which means that the boundaries identified in the random effects surface can also be interpreted as boundaries in the risk surface. The model summary table displayed above has the same format as the previous example with two exceptions. Firstly, because only a single Markov chain is generated in this example, the potential scale reduction factor (PSRF) cannot be computed, so instead the convergence Z-score diagnostic proposed by Geweke (1992) (`Geweke.diag`) is presented. Secondly, the model summary also contains a column in the parameter summary table headed `alpha.min`, which only applies to the dissimilarity metrics and is hence `NA` for the remaining parameters. The value of `alpha.min` is the threshold value for the regression parameter α, below which the dissimilarity metric has no effect in identifying boundaries in the response (random effects) surface. A brief description is given in Section 2.1, while full details are given in Lee and Mitchell (2012). For these data the posterior mean and 95% credible interval lie completely above this threshold, suggesting that the income deprivation dissimilarity metric has identified a number of boundaries.

The number and locations of these boundaries are summarised in the element of the output list called `chain1$localised.structure$W.posterior`, which is a $K \times K$ symmetric matrix containing the posterior median for the set $\{w_{kj}|k \sim j\}$, where $k \sim j$ denotes that areas $(\mathcal{S}_k, \mathcal{S}_j)$ share a common border. Values equal to zero represent a boundary, values equal to one correspond to no boundary, while `NA` values correspond to non-adjacent areas. The locations of these boundaries can be overlaid on a map of the estimated disease risk (that

is the posterior mean of $\{\theta_k\}$). This is done in two steps, the first being the creation of the boundaries as a `SpatialPoints` object using the following code.

```
R> border.locations <- chain1$localised.structure$W.posterior
R> respiratorydata.sf$risk <- chain1$fitted.values /
+                                        respiratorydata.sf$expected
R> boundary.final <- highlight.borders(border.locations=border.locations,
+                          sfdata=respiratorydata.sf)
R> st_crs(boundary.final) <- '+proj=longlat +datum=WGS84 +no_defs'
```

The first line saves the matrix of border locations, while the second adds the estimated risk values to the `respiratorydata.sf` object. The next line identifies the boundary points using the **CARBayes** function `highlight.borders()` and formats them to enable plotting. Then the final line specifies that this object has a longitude and latitude coordinate reference system. Then plotting can be done using the code below, and the result is presented in Figure 5.

```
R> map3 <- mapview(respiratorydata.sf, zcol = "risk",
+                 col.regions=brewer.pal(9, "YlOrRd"), alpha.regions=0.6,
+                 layer.name="Risk", lwd=0, homebutton=FALSE) +
+           mapview(boundary.final, color="blue", alpha.regions=1,  lwd=3,
+                 legend=FALSE, homebutton=FALSE)
R> removeMapJunk(map3, junk = c("zoomControl", "layersControl"))
```

The figure shows the estimated risk surface and the locations of the boundaries (denoted by blue dots). The model has identified 99 boundaries in the risk surface. The majority of these visually correspond to sizeable changes in the risk surface, suggesting that the model has the power to distinguish between boundaries and non-boundaries. The notable boundaries are the demarcation between the low risk (shaded yellow) city centre / west end of Glasgow in the middle of the region and the deprived neighbouring areas on both sides (shaded orange and red), which include Easterhouse / Parkhead in the east and Knightswood / Drumchapel in the west. The other interesting feature of this map is that the boundaries are not closed, suggesting that the spatial pattern in risk is more complex than being partitioned into groups of non-overlapping areas of similar risk.

6. Discussion

This vignette has illustrated the R package **CARBayes**, which can fit a number of commonly used conditional autoregressive models to spatial areal unit data, as well as the localised spatial smoothing models proposed by Lee and Mitchell (2012) and Lee and Sarran (2015). The response data can be binomial, Gaussian, multinomial, Poisson or ZIP, with link functions logit, identity, logit, natural log and (natural log / logit) respectively. The availability of areal unit data has grown dramatically in recent times, due to the launch of freely available on-line databases such as http://statistics.gov.scot. This increased availability of spatial data has fuelled a growth of modelling in this area, leading to the need for user friendly software such as **CARBayes** and **CARBayesST** (for spatio-temporal modelling) for use by both statisticians and non-statisticians alike. Future development of this package will increase

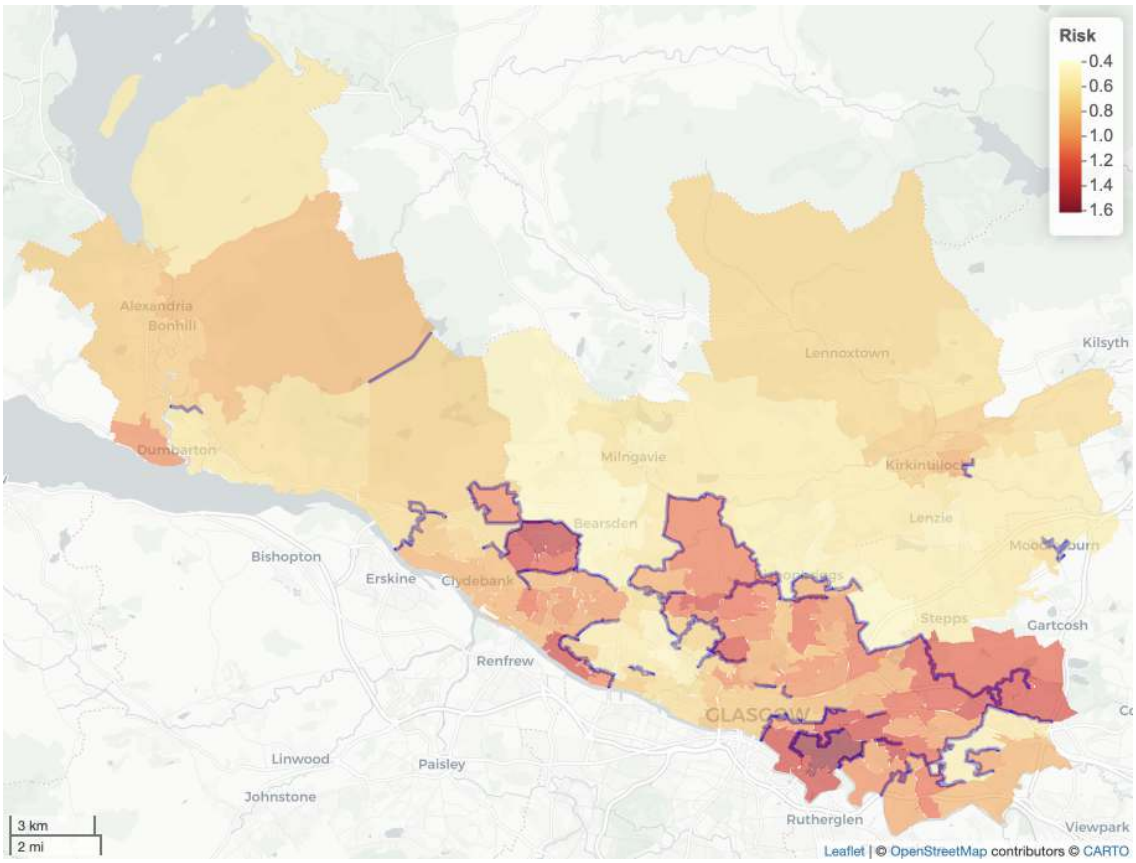

Figure 5: Map displaying the estimated risk and the locations of the boundaries for the northern part of Greater Glasgow.

the number of spatial models available, particularly those with more complex multivariate and multilevel structures.

Acknowledgements

The data and shapefiles used in sections 5 and 6 of this vignette were provided by the Scottish Government.

References

Appelhans T, Detsch F, Reudenbach C, Woellauer S (2022). *mapview: Interactive Viewing of Spatial Data in R*. R package version 2.11.0, URL https://CRAN.R-project.org/package=mapview.

Besag J, Higdon D (1999). "Bayesian Analysis of Agricultural Field Experiments." *Journal of the Royal Statistical Society Series B*, **61**, 691–746.

Besag J, York J, Mollié A (1991). "Bayesian Image Restoration with Two Applications in Spatial Statistics." *Annals of the Institute of Statistics and Mathematics*, **43**, 1–59.

Bivand R, Pebesma E, Gomez-Rubio V (2013). *Applied spatial data analysis with R*. 2nd edition. Springer, NY.

Brewer M, Nolan A (2007). "Variable Smoothing in Bayesian Intrinsic Autoregressions." *Environmetrics*, **18**, 841–857.

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

Congdon P (2005). *Bayesian models for categorical data*. 1st edition. John Wiley and 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/.

Gavin J, Jennison C (1997). "A subpixel Image Restoration Algorithm." *Journal of Computational and Graphical and Statistics*, **6**, 182–201.

Gelfand A, Vounatsou P (2003). "Proper multivariate conditional autoregressive models for spatial data analysis." *Biostatistics*, **4**, 11–25.

Gelman A, Carlin J, Stern H, Rubin D (2003). *Bayesian Data Analysis*. 2nd edition. Chapman and Hall/CRC, London.

Geweke J (1992). "Evaluating the Accuracy of Sampling-Based Approaches to the Calculation of Posterior Moments." In *IN BAYESIAN STATISTICS*, pp. 169–193. University Press.

Green P, Richardson S (2002). "Hidden Markov Models and Disease Mapping." *Journal of the American Statistical Association*, **97**, 1055–1070.

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

Kavanagh L, Lee D, Pryce G (2016). "Is Poverty Decentralising? Quantifying Uncertainty in the Decentralisation of Urban Poverty." *Annals of the American Association of Geographers*, **106**, 1286–1298.

Lee D (2011). "A Comparison of Conditional Autoregressive Models Used in Bayesian Disease Mapping." *Spatial and Spatio-temporal Epidemiology*, **2**, 79–89.

Lee D (2013). "CARBayes: An R Package for Bayesian Spatial Modeling with Conditional Autoregressive Priors." *Journal of Statistical Software*, **55**, 13.

Lee D (2022). *CARBayesdata: Data Sets Used in the Vignette Accompanying the CARBayes Package*. R package version 3.0, URL https://CRAN.R-project.org/package=CARBayesdata.

Lee D, Ferguson C, Mitchell R (2009). "Air Pollution and Health in Scotland: A Multicity Study." *Biostatistics*, **10**, 409–423.

Lee D, Mitchell R (2012). "Boundary Detection in Disease Mapping Studies." *Biostatistics*, **13**, 415–426.

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.

Lee D, Sarran C (2015). "Controlling for unmeasured confounding and spatial misalignment in long-term air pollution and health studies." *Environmetrics*, **26**, 477–487.

Leroux B, Lei X, Breslow N (2000). "Estimation of Disease Rates in Small Areas: A New Mixed Model for Spatial Dependence." In M Halloran, D Berry (eds.), *Statistical Models in Epidemiology, the Environment and Clinical Trials*, pp. 179–191. Springer-Verlag, New York.

Lu H, Reilly C, Banerjee S, Carlin B (2007). "Bayesian Areal Wombling Via Adjacency Modelling." *Environmental and Ecological Statistics*, **14**, 433–452.

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

Ma H, Carlin B (2007). "Bayesian Multivariate Areal Wombling for Multiple Disease Boundary Analysis." *Bayesian Analysis*, **2**, 281–302.

Martin AD, Quinn KM, Park JH (2011). "MCMCpack: Markov Chain Monte Carlo in R." *Journal of Statistical Software*, **42**(9), 22. URL http://www.jstatsoft.org/v42/i09/.

Moran P (1950). "Notes on continuous stochastic phenomena." *Biometrika*, **37**, 17–23, DOI:10.1093/biomet/37.1–2.17.

Neuwirth E (2022). *RColorBrewer: ColorBrewer Palettes*. R package version 1.1-3, URL https://CRAN.R-project.org/package=RColorBrewer.

Novomestky F (2012). *matrixcalc: Collection of functions for matrix calculations*. R package version 1.0-3, URL https://CRAN.R-project.org/package=matrixcalc.

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/.

Roberts G, Rosenthal J (1998). "Optimal scaling of discrete approximations to the Langevin diffusions." *Journal of the Royal Statistical Society Series 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 Series B*, **71**, 319–392.

Spiegelhalter D, Best N, Carlin B, Van der Linde A (2002). "Bayesian Measures of Model Complexity and Fit." *Journal of the Royal Statistical Society series 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.

Ugarte M, Ibanez B, Militino A (2004). "Testing for Poisson Zero Inflation in Disese Mapping." *Biometrical Journal*, **46**, 526–539.

Venables WN, Ripley BD (2002). *Modern Applied Statistics with S*. Fourth edition. Springer, New York. ISBN 0-387-95457-0, URL http://www.stats.ox.ac.uk/pub/MASS4.

Wakefield J (2007). "Disease Mapping and Spatial Regression with Count Data." *Biostatistics*, **8**, 158–183.

Wall M (2004). "A Close Look at the Spatial Structure Implied by the CAR and SAR Models." *Journal of Statistical Planning and Inference*, **121**, 311–324.

Watanabe S (2010). "Asymptotic equivalence of the Bayes cross validation and widely applicable information criterion in singular learning theory." *Journal of Machine Learning Research*, **11**, 3571–3594.

Womble W (1951). "Differential Systematics." *Science*, **114**, 315–322.

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/