

STK4150/9150 - Environmental and Spatial Statistics

Spatial lattice processes, Part II

Thordis Thorarinsdottir

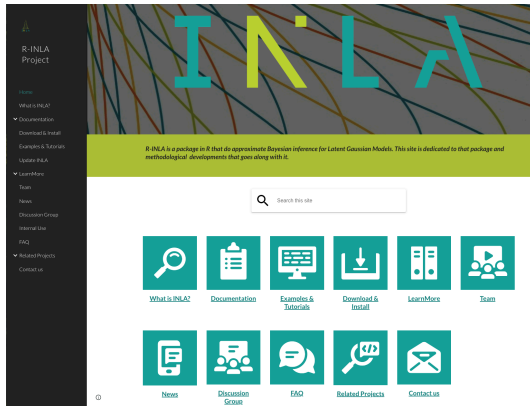
Spring 2026

Statistical inference for non-Gaussian data with latent RFs

Whether we are interested in inference on the data themselves, or the latent random field, the inference is generally complicated if non-Gaussian data are involved.

Often, the inference requires a hierarchical modelling set-up which is easiest to solve in a Bayesian setting.

Several sophisticated inference frameworks exist; we will focus on a fairly simple set-up.



Stan is a state-of-the-art platform for statistical modeling and high-performance statistical computation. Thousands of users rely on Stan for statistical modeling, data analysis, and prediction in the social, biological, and physical sciences, engineering, and business.

Stan interfaces with the most popular data analysis languages (R, Python, shell, MATLAB, Julia, Stata) and runs on all major platforms (Linux, Mac, Windows). To get started using Stan begin with the [Installation](#) and [Documentation](#) pages.

Users specify log density functions in the Stan probabilistic programming language and then fit the models to data using:

- full Bayesian statistical inference with MCMC sampling (NUTS, HMC)
- approximate Bayesian inference with variational inference (ADVI, Pathfinder)
- penalized maximum likelihood estimation with optimization (L-BFGS)

Stan's math library provides differentiable probability functions & linear algebra (C++ autodiff). Additional R packages provide expression-based linear modeling, posterior visualization, and leave-one-out cross-validation.

Follow us on Mastodon!

Package ‘CARBayes’

November 17, 2023

Type Package

Title Spatial Generalised Linear Mixed Models for Areal Unit Data

Version 6.1

Date 2023-11-17

Author Duncan Lee

Maintainer Duncan Lee <Duncan.Lee@glasgow.ac.uk>

Description Implements a class of 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, including the BYM model (Besag et al., 1991, <doi:10.1007/BF00116466>) and Leroux model (Leroux et al., 2000, <doi:10.1007/978-1-4612-1284-3_4>). 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. Full details are given in the vignette accompanying this package. The initial creation of this package was supported by the Economic and Social Research Council (ESRC) grant RES-000-22-4256, and ongoing 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.

License GPL (>= 2)

Depends MASS, R (>= 3.5.0), Rcpp (>= 0.11.5)

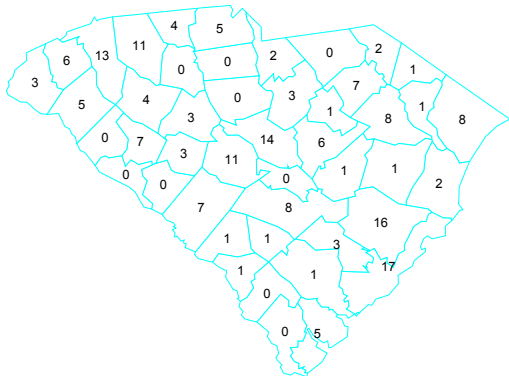
Imports CARBayesdata, coda, dplyr, GGally, glmnet, igraph, mapview, MCMCpack, parallel, RColorBrewer, sf, spam, sspdep, stats,

Congenital abnormality deaths in South Carolina in 1990

- Number of congenital abnormality deaths per county
- Also: Expected rate, poverty and income

```
head(SCcg90)
```

	obs	expe	pov	inc	name
1	0	1.0807	13.6	36.786	Abbeville
2	7	6.3775	13.8	38.534	Aiken
3	1	0.6220	32.3	20.727	Allendale
4	5	6.6854	11.0	37.205	Anderson
5	1	0.9142	24.2	24.300	Bamberg
6	1	1.0744	19.9	27.607	Barnwell



```
library(sf)
library(tigris)
options(tigris_use_cache = TRUE)

# Read in SC shapefile with tigris
sc_counties_sf <- counties(state = "SC", cb = TRUE, class = "sf")
head(sc_counties_sf[,c(6,9,13)],3)
```

Simple feature collection with 3 features and 2 fields

Geometry type: MULTIPOLYGON

Dimension: XY

Bounding box: xmin: -81.49228 ymin: 32.07689 xmax: -78.54732 ymax: 35.16489

Geodetic CRS: NAD83

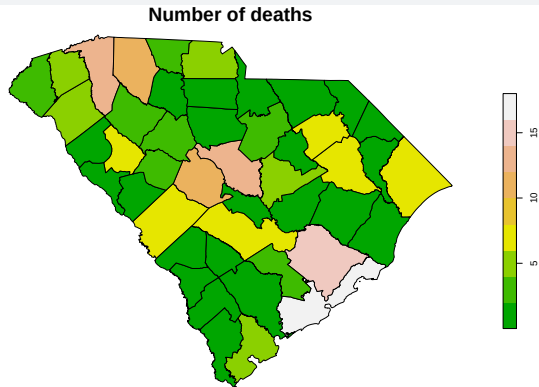
	NAME	STATE_NAME	geometry
114	Beaufort	South Carolina	MULTIPOLYGON (((-81.01611 3...
115	Horry	South Carolina	MULTIPOLYGON (((-79.33884 3...
116	York	South Carolina	MULTIPOLYGON (((-81.49228 3...

```
library(dplyr)

# Prepare names for matching:
#   make lowercase and trim whitespace
sc_counties_sf <- sc_counties_sf %>%
  mutate(NAME_clean = tolower(trimws(NAME)))
SCcg90 <- SCcg90 %>%
  mutate(name_clean = tolower(trimws(name)))

# Join the two data data sets by county name
sc_sf <- sc_counties_sf %>%
  left_join(SCcg90,
            by = c("NAME_clean" = "name_clean"))
```

```
plot(sc_sf["obs"], pal=terrain.colors,
     main="Number of deaths", cex.main=2)
```



Next, we create the neighborhood structures:

- `poly2nb()` creates a neighbor list from polygons based on shared boundaries
- `nb2mat()` uses the neighbor list to create a spatial weights matrix

```
library(spdep)
```

```
nb <- poly2nb(sc_sf, queen = TRUE)
summary(nb)
```

Neighbour list object:

Number of regions: 46

Number of nonzero links: 226

Percentage nonzero weights: 10.68053

Average number of links: 4.913043

Link number distribution:

```
 2  3  4  5  6  7  8  9
2  5 12 13  7  5  1  1
```

2 least connected regions:

38 41 with 2 links

1 most connected region:

17 with 9 links

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

```
W.mat[9:11, 37:40]
```

```
      37 38 39 40
9      0  0  0  1
10     1  0  1  0
11     0  0  0  1
```

Poisson data model with Besag-York-Mollie CAR model

$$Z_k | \lambda_k \sim \text{Poisson}(\lambda_k) \quad \text{for } k = 1, \dots, K (\text{data model})$$

$$\ln(\lambda_k) = \mathbf{x}_k^\top \boldsymbol{\beta} + O_k + \psi_k \quad (O_k \text{ known offset})$$

$$\boldsymbol{\beta} \sim \text{Gau}(\boldsymbol{\mu}_\beta, \boldsymbol{\Sigma}_\beta)$$

$$\psi_k = \varphi_k + \theta_k$$

$$\varphi_k | \boldsymbol{\varphi}_{-k}, \mathbf{W}, \tau^2 \sim \text{Gau}\left(\frac{\sum_{i=1}^K w_{ki} \varphi_i}{\sum_{i=1}^K w_{ki}}, \frac{\tau^2}{\sum_{i=1}^K w_{ki}}\right) \quad (\text{spatial random effects})$$

$$\theta_k \sim \text{Gau}(0, \sigma^2) \quad (\text{independent random effects})$$

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

```
library(CARBayes)
form <- obs ~ 1 + offset(log(expe))

model.spatial <- S.CARbym(formula=form, family="poisson", data=sc_sf, W=W.mat,
                          burnin=10000, n.sample=110000)

print(model.spatial)
```

```
#####
#### Model fitted
#####
Likelihood model - Poisson (log link function)
Random effects model - BYM CAR
Regression equation - obs ~ 1 + offset(log(expe))

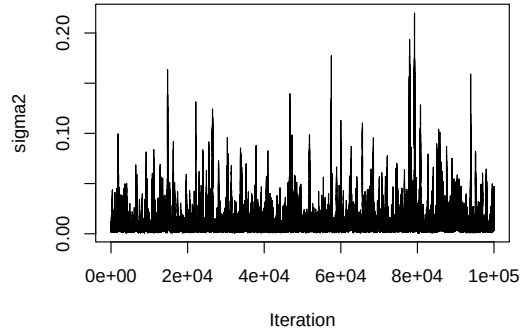
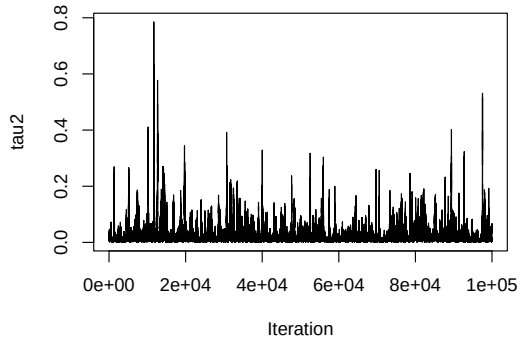
#####
#### MCMC details
#####
Total number of post burnin and thinned MCMC samples generated - 1e+05
Number of MCMC chains used - 1
Length of the burnin period used for each chain - 10000
Amount of thinning used - 1

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

      Mean      2.5%  97.5% n.effective Geweke.diag
(Intercept) 0.0360 -0.1142 0.1783      14109.3        0.5
tau2         0.0186  0.0025 0.0861        974.2       -0.8
sigma2       0.0098  0.0018 0.0394        889.8       -1.6

DIC = 170.3151      p.d = 3.59631      LMPL = -85.05
```

```
par(mfrow=c(1,2))  
plot(1:100000, model.spatial$samples$tau2, type="l", xlab="Iteration", ylab="tau2")  
plot(1:100000, model.spatial$samples$sigma2, type="l", xlab="Iteration", ylab="sigma2")
```



```
form <- obs ~ 1 + pov + inc + offset(log(expe))  
  
model.spatial2 <- S.CARbym(formula=form, family="poisson", data=sc_sf, W=W.mat,  
                           burnin=10000, n.sample=110000)  
  
print(model.spatial2)
```

```
#####
#### Model fitted
#####
Likelihood model - Poisson (log link function)
Random effects model - BYM CAR
Regression equation - obs ~ 1 + pov + inc + offset(log(expe))

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

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

      Mean    2.5%  97.5% n.effective Geweke.diag
(Intercept) 0.6249 -2.3151 3.5287      1824.2      -0.4
pov          -0.0162 -0.0885 0.0548      1696.8       0.5
inc          -0.0097 -0.0630 0.0442      2025.3       0.3
tau2         0.0242  0.0026 0.1290       840.7      -0.4
sigma2       0.0102  0.0019 0.0403       856.2     -1.5

DIC = 173.7781      p.d = 5.592328      LMPL = -86.82
```

References

Duncan, E. W., S. M. Cramb, P. D. Baade, K. L. Mengersen, T. Saunders, and J. F. Aitken. 2019. Developing a Cancer Atlas using Bayesian Methods: A Practical Guide for Application and Interpretation. Brisbane: Queensland University of Technology (QUT) and Cancer Council Queensland. Available at <https://atlas.cancer.org.au/developing-a-cancer-atlas/> (Accessed on February 28, 2024).

Lawson, A. 2019. Bayesian DM code examples. Available at <https://github.com/Andrew9Lawson/Bayesian-DM-code-examples/tree/master> (Accessed on February 28, 2024)

Lee, D. 2024. Package "CARBayes" (manual) and Vignette for CARBayes package. Available at <https://cran.r-project.org/web/packages/CARBayes/index.html> (Accessed on February 7, 2026)

Pebesma, E., R. Bivand, E. Racine, M. Sumner, I. Cook, T. Keitt, R. Lovelace, H. Wickham, J. Ooms, K. Müller, T.L. Pedersen, D. Baston and D. Dunnington. 2026. Package "sf" (manual) and Vignettes for sf package. Available at <https://cran.r-project.org/web/packages/sf/index.html> (Accessed on February 7, 2026)