

# Package ‘CARBayes’

June 1, 2017

**Type** Package

**Title** Spatial Generalised Linear Mixed Models for Areal Unit Data

**Version** 5.0

**Date** 2017-06-01

**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. The response variable can be binomial, Gaussian or Poisson, 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 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 on-going development has / is supported by the Engineering and Physical Science Research Council (EPSRC) grant EP/J017442/1, ESRC grant ES/K006460/1, and Innovate UK / Natural Environment Research Council (NERC) grant NE/N007352/1.

**License** GPL (>= 2)

**Depends** MASS, R (>= 3.0.0), Rcpp (>= 0.11.5)

**Imports** CARBayesdata, coda, matrixcalc, MCMCpack, sp, spam, spdep, stats, truncnorm, utils

**Suggests** boot, deldir, foreign, grid, maptools, Matrix, nlme, shapefiles, splines

**LinkingTo** Rcpp

**LazyLoad** yes

**ByteCompile** yes

**URL** <http://github.com/duncanplee/CARBayes>

**BugReports** <http://github.com/duncanplee/CARBayes/issues>

**NeedsCompilation** yes

**Repository** CRAN

**Date/Publication** 2017-06-01 10:58:32 UTC

## R topics documented:

CARBayes-package . . . . .	2
coef.CARBayes . . . . .	4
combine.data.shapefile . . . . .	4
fitted.CARBayes . . . . .	5
highlight.borders . . . . .	6
logLik.CARBayes . . . . .	7
model.matrix.CARBayes . . . . .	7
MVS.CARleroux . . . . .	8
print.CARBayes . . . . .	11
residuals.CARBayes . . . . .	12
S.CARbym . . . . .	12
S.CARdissimilarity . . . . .	15
S.CARleroux . . . . .	17
S.CARlocalised . . . . .	20
S.CARmultilevel . . . . .	23
S.glm . . . . .	26
summarise.lincomb . . . . .	28
summarise.samples . . . . .	29
<b>Index</b>	<b>30</b>

---

CARBayes-package

*Spatial Generalised Linear Mixed Models for Areal Unit Data*

---

## 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. The response variable can be binomial, Gaussian or Poisson, 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 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 on-going development has / is supported by the Engineering and Physical Science Research Council (EPSRC) grant EP/J017442/1, ESRC grant ES/K006460/1, and Innovate UK / Natural Environment Research Council (NERC) grant NE/N007352/1. Version 5.0 has a number of changes from version 4.7, including: (i) inclusion of a simple model with no random effects (S.CARglm()); (ii) inclusion of a two-level hierarchical model for modelling individuals within areas (S.CARmultilevel()); (iii) an argument in

the model fitting functions to allow the random effects to be updated using either random walk Metropolis steps or the Metropolis adjusted Langevin algorithm (MALA); and (iv) calculation of the percentage of the deviance explained by the model.

## Details

Package: CARBayes  
Type: Package  
Version: 5.0  
Date: 2017-06-01  
License: GPL (>= 2)

## Author(s)

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

## References

- Besag, J. and York, J and Mollie, A (1991). Bayesian image restoration with two applications in spatial statistics. *Annals of the Institute of Statistics and Mathematics* 43, 1-59.
- Gelfand, A and Vounatsou, P (2003). Proper multivariate conditional autoregressive models for spatial data analysis, *Biostatistics*, 4, 11-25.
- Kavanagh, L., D. Lee, and G. Pryce (2016). Is Poverty Decentralising? Quantifying Uncertainty in the Decentralisation of Urban Poverty, *Annals of the American Association of Geographers*, 106, 1286-1298.
- Lee, D. and Mitchell, R (2012). Boundary detection in disease mapping studies. *Biostatistics*, 13, 415-426.
- Lee, D and 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 and Breslow, N (2000). Estimation of disease rates in small areas: A new mixed model for spatial dependence, Chapter *Statistical Models in Epidemiology, the Environment and Clinical Trials*, Halloran, M and Berry, D (eds), pp. 135-178. Springer-Verlag, New York.
- Roberts, G and Rosenthal, J (1998). Optimal scaling of discrete approximations to the Langevin diffusions, *Journal of the Royal Statistical Society Series B* 60, 255-268.

## Examples

```
## See the examples in the function specific help files and in the vignette  
## accompanying this package.
```

---

coef.CARBayes	<i>Extract the regression coefficients from a model.</i>
---------------	--

---

### Description

This function takes a CARBayes object and returns the vector of estimated regression coefficients.

### Usage

```
## S3 method for class 'CARBayes'  
coef(object, ...)
```

### Arguments

object	A CARBayes fitted model object.
...	Ignored.

### Author(s)

Duncan Lee

---

combine.data.shapefile	<i>Combines a data frame with a shapefile to create a SpatialPolygons-DataFrame object.</i>
------------------------	---

---

### Description

This function combines a data frame with a shapefile to create a SpatialPolygonsDataFrame object from the 'sp' package. The creation of this object allows the variables in the data frame to be mapped using the 'splot()' function, and the neighbourhood matrix  $W$  to be created using the 'poly2nb' and 'nb2mat' functions. An example is given in the vignette accompanying this package. The mapping of the data to the shapefile is done by matching the rownames of the data frame to the first column in the dbf file.

### Usage

```
combine.data.shapefile(data, shp, dbf)
```

**Arguments**

data	A data frame containing the variables relating to the K areas you wish to map or model. The row names of this data frame must appear in the first column of the dbf file.
shp	The .shp part of a shapefile containing the polygons for each of the K areas that the data relate to.
dbf	The .dbf part of the shapefile containing a lookup table whose first column should include the K row names of the data frame.

**Value**

A SpatialPolygonsDataFrame object from the 'sp' package containing the combined data and shapefile object.

**Author(s)**

Duncan Lee

**Examples**

```
## See the vignette accompanying this package for an example of its use.
```

---

fitted.CARBayes	<i>Extract the fitted values from a model.</i>
-----------------	--

---

**Description**

This function takes a CARBayes object and returns the vector of fitted values based on posterior medians.

**Usage**

```
## S3 method for class 'CARBayes'  
fitted(object, ...)
```

**Arguments**

object	A CARBayes fitted model object.
...	Ignored.

**Author(s)**

Duncan Lee

---

highlight.borders	<i>Creates a SpatialPoints object identifying a subset of borders between neighbouring areas.</i>
-------------------	---

---

### Description

Creates a SpatialPoints object identifying a subset of borders between neighbouring areas, which allows them to be overlaid on a map. An example is given in the vignette accompanying this package.

### Usage

```
highlight.borders(border.locations, spdata)
```

### Arguments

border.locations	A K by K matrix, where K is the number of areas, containing 3 distinct values: NA for non-neighbouring areas; 0 for borders between neighbouring areas to be highlighted on a map; and 1 for borders between neighbouring areas not to be highlighted on a map.
spdata	The 'SpatialPolygonsDataFrame' object used for plotting the data and creating the original neighbourhood matrix W.

### Value

A 'SpatialPoints' object from the 'sp' package, which contains the vertices of all the borders to be highlighted on the map. The mapping can be done using the 'spplot()' function, see the vignette accompanying this package for an example.

### Author(s)

Duncan Lee

### Examples

```
## See the vignette accompanying this package for an example of its use.
```

---

logLik.CARBayes	<i>Extract the estimated loglikelihood from a fitted model.</i>
-----------------	---

---

**Description**

This function takes a CARBayes object and returns the estimated loglikelihood.

**Usage**

```
## S3 method for class 'CARBayes'  
logLik(object, ...)
```

**Arguments**

object	A CARBayes fitted model object.
...	Ignored.

**Author(s)**

Duncan Lee

---

model.matrix.CARBayes	<i>Extract the Model (design) matrix from a model.</i>
-----------------------	--

---

**Description**

This function takes a CARBayes object and returns the design matrix.

**Usage**

```
## S3 method for class 'CARBayes'  
model.matrix(object, ...)
```

**Arguments**

object	A CARBayes fitted model object.
...	Ignored.

**Author(s)**

Duncan Lee

---

MVS.CARleroux	<i>Fit a multivariate spatial generalised linear mixed model to data, where the random effects are modelled by a multivariate conditional autoregressive model.</i>
---------------	---

---

## Description

Fit a multivariate spatial generalised linear mixed model to areal unit data, where the response variable can be binomial or Poisson. The linear predictor is modelled by known covariates and a vector of random effects. The latter account for both spatial and between variable correlation, via a Kronecker product formulation. Spatial correlation is captured by the conditional autoregressive (CAR) prior proposed by Leroux et al. (1999), and between variable correlation is captured by a between variable covariance matrix with no fixed structure. This is a type of multivariate conditional autoregressive (MCAR). Further details are given in the vignette accompanying this package. Independent (over space) random effects can be obtained by setting (fix.rho=TRUE, rho=0), in which case the neighbourhood matrix  $W$  is not part of the model. In this case enter a fake  $W$  matrix that is a  $K$  by  $K$  matrix of zeros, where  $K$  is the number of spatial units. Similarly, the intrinsic MCAR model can be obtained by setting (fix.rho=TRUE, rho=1). Inference is conducted in a Bayesian setting using Markov chain Monte Carlo (MCMC) simulation. Missing (NA) values are allowed in the response, and posterior predictive distributions are created for the missing values for predictive purposes. These are saved in the 'samples' argument in the output of the function and are denoted by 'Y'.

## Usage

```
MVS.CARleroux(formula, family, data=NULL, trials=NULL, W, burnin,
n.sample, thin=1, prior.mean.beta=NULL, prior.var.beta=NULL,
prior.Sigma.df=NULL, prior.Sigma.scale=NULL, fix.rho=FALSE, rho=NULL, MALA=TRUE,
verbose=TRUE)
```

## Arguments

formula	A formula for the covariate part of the model using the syntax of the <code>lm()</code> function. Offsets can be included here using the <code>offset()</code> function. The response and each covariate should be vectors of length $(KJ)*1$ , where $K$ is the number of spatial units and $J$ is the number of different variables. Each vector should be ordered so that the first $J$ data points are the set of all $J$ variables for spatial location 1, the next $J$ are the set of variables for spatial location 2 and so on. Missing (NA) values are allowed in the response.
family	One of either 'binomial' or 'poisson', which respectively specify a binomial likelihood model with a logistic link function, or a Poisson likelihood model with a log link function.
data	An optional data.frame containing the variables in the formula.
trials	A vector the same length as the response containing the total number of trials for each area. Only used if family='binomial'.

<code>W</code>	A non-negative K by K neighbourhood matrix (where K is the number of spatial units). Typically a binary specification is used, where the jkth element equals one if areas (j, k) are spatially close (e.g. share a common border) and is zero otherwise. The matrix can be non-binary, but each row must contain at least one non-zero entry.
<code>burnin</code>	The number of MCMC samples to discard as the burn-in period.
<code>n.sample</code>	The number of MCMC samples to generate.
<code>thin</code>	The level of thinning to apply to the MCMC samples to reduce their temporal autocorrelation. Defaults to 1 (no thinning).
<code>prior.mean.beta</code>	A vector of prior means for the regression parameters beta (Gaussian priors are assumed). Defaults to a vector of zeros.
<code>prior.var.beta</code>	A vector of prior variances for the regression parameters beta (Gaussian priors are assumed). Defaults to a vector with values 100000.
<code>prior.Sigma.df</code>	The prior degrees of freedom for the Inverse-Wishart prior for Sigma. Defaults to J+1.
<code>prior.Sigma.scale</code>	The prior J times J scale matrix for the Inverse-Wishart prior for Sigma. Defaults to the identity matrix.
<code>fix.rho</code>	Logical, should the spatial dependence parameter rho be fixed rather than being estimated in the model.
<code>rho</code>	The value that the spatial dependence parameter is set to if <code>fix.rho=TRUE</code> . Must be in the interval [0, 1], where 0 corresponds to independence while 1 corresponds to fitting the intrinsic CAR prior. Note, if <code>rho=0</code> then the W matrix is not included in the model so enter a K by K matrix of zeros instead.
<code>MALA</code>	Logical, should the function use MALA (TRUE, default) or simple random walk (FALSE) updates for the random effects. Not applicable if <code>family='gaussian'</code> .
<code>verbose</code>	Logical, should the function update the user on its progress.

## Value

<code>summary.results</code>	A summary table of the parameters.
<code>samples</code>	A list containing the MCMC samples from the model.
<code>fitted.values</code>	A vector of fitted values for each area and response variable in the same order as the data vectors.
<code>residuals</code>	A matrix with 3 columns where each column is a type of residual and each row relates to an area. The types are "response" (raw), "pearson", and "deviance".
<code>modelfit</code>	Model fit criteria including the Deviance Information Criterion (DIC) and its corresponding estimated effective number of parameters (p.d), the Log Marginal Predictive Likelihood (LMPL), the Watanabe-Akaike Information Criterion (WAIC) and its corresponding estimated number of effective parameters (p.w), the log-likelihood, and the percentage deviance explained.
<code>accept</code>	The acceptance probabilities for the parameters.

```

localised.structure
                NULL, for compatability with the other models.
formula         The formula for the covariate and offset part of the model.
model           A text string describing the model fit.
X               The design matrix of covariates.

```

### Author(s)

Duncan Lee

### References

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

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

Leroux, B., X. Lei, and N. Breslow (2000). Estimation of disease rates in small areas: A new mixed model for spatial dependence, Chapter *Statistical Models in Epidemiology, the Environment and Clinical Trials*, Halloran, M and Berry, D (eds), pp. 135-178. Springer-Verlag, New York.

### Examples

```

#####
#### Run the model on simulated data on a lattice
#####
#### Load other libraries required
library(MASS)

#### Set up a square lattice region
x.easting <- 1:10
x.northing <- 1:10
Grid <- expand.grid(x.easting, x.northing)
K <- nrow(Grid)

#### set up distance and neighbourhood (W, based on sharing a common border) matrices
distance <- as.matrix(dist(Grid))
W <- array(0, c(K,K))
W[distance==1] <-1
K <- nrow(W)

#### Generate the correlation structures
Q.W <- 0.99 * (diag(apply(W, 2, sum)) - W) + 0.01 * diag(rep(1,K))
Q.W.inv <- solve(Q.W)

Sigma <- matrix(c(1,0.5,0, 0.5,1,0.3, 0, 0.3, 1), nrow=3)
Sigma.inv <- solve(Sigma)
J <- nrow(Sigma)

precision.phi <- kronecker(Q.W, Sigma.inv)

```

```
var.phi <- solve(precision.phi)

#### Generate data
N.all <- K * J
x1 <- rnorm(N.all)
x2 <- rnorm(N.all)
phi <- mvrnorm(n=1, mu=rep(0,N.all), Sigma=var.phi)

lp <- 0.1 * x1 - 0.1 * x2 + phi
prob <- exp(lp) / (1 + exp(lp))
trials <- rep(100,N.all)
Y <- rbinom(n=N.all, size=trials, prob=prob)

#### Run the Leroux model
formula <- Y ~ x1 + x2
## Not run: model <- MVS.CARleroux(formula=formula, family="binomial",
trials=trials, W=W, burnin=20000, n.sample=100000)
## End(Not run)
```

---

print.CARBayes

*Print a summary of a fitted CARBayes model to the screen.*

---

## Description

This function takes a CARBayes object and returns a summary of the fitted model. The summary includes posterior medians and 95 percent credible intervals, the number of samples, the acceptance rate, the effective number of independent samples and the Geweke convergence diagnostic in the form of a Z-score.

## Usage

```
## S3 method for class 'CARBayes'
print(x, ...)
```

## Arguments

x	A 'CARBayes' fitted model object
...	Ignored

## Author(s)

Duncan Lee

---

residuals.CARBayes	<i>Extract the residuals from a model.</i>
--------------------	--

---

### Description

This function takes a CARBayes object and returns a set of residuals. The allowable types of residual are "response" (raw), "pearson", and "deviance".

### Usage

```
## S3 method for class 'CARBayes'
residuals(object, type, ...)
```

### Arguments

object	A CARBayes fitted model object.
type	A text string and one of c("response", "pearson", "deviance").
...	Ignored.

### Author(s)

Duncan Lee

---

S.CARbym	<i>Fit a spatial generalised linear mixed model to data, where the random effects have a BYM conditional autoregressive prior.</i>
----------	--

---

### Description

Fit a spatial generalised linear mixed model to areal unit data, where the response variable can be binomial, or Poisson. Note, a Gaussian likelihood is not allowed because of a lack of identifiability among the parameters. The linear predictor is modelled by known covariates and a vector of random effects. The latter are modelled by the BYM conditional autoregressive prior proposed by Besag et al. (1991), and further details are given in the vignette accompanying this package. Inference is conducted in a Bayesian setting using Markov chain Monte Carlo (MCMC) simulation. Missing (NA) values are allowed in the response, and posterior predictive distributions are created for the missing values for predictive purposes. These are saved in the 'samples' argument in the output of the function and are denoted by 'Y'.

### Usage

```
S.CARbym(formula, family, data=NULL, trials=NULL, W, burnin, n.sample,
thin=1, prior.mean.beta=NULL, prior.var.beta=NULL, prior.tau2=NULL,
prior.sigma2=NULL, MALA=TRUE, verbose=TRUE)
```

**Arguments**

formula	A formula for the covariate part of the model using the syntax of the <code>lm()</code> function. Offsets can be included here using the <code>offset()</code> function. The response can contain missing (NA) values.
family	One of either 'binomial' or 'poisson', which respectively specify a binomial likelihood model with a logistic link function, or a Poisson likelihood model with a log link function.
data	An optional data.frame containing the variables in the formula.
trials	A vector the same length as the response containing the total number of trials for each area. Only used if <code>family='binomial'</code> .
W	A non-negative K by K neighbourhood matrix (where K is the number of spatial units). Typically a binary specification is used, where the jkth element equals one if areas (j, k) are spatially close (e.g. share a common border) and is zero otherwise. The matrix can be non-binary, but each row must contain at least one non-zero entry.
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).
prior.mean.beta	A vector of prior means for the regression parameters beta (Gaussian priors are assumed). Defaults to a vector of zeros.
prior.var.beta	A vector of prior variances for the regression parameters beta (Gaussian priors are assumed). Defaults to a vector with values 100000.
prior.tau2	The prior shape and scale in the form of <code>c(shape, scale)</code> for an Inverse-Gamma(shape, scale) prior for tau2. Defaults to <code>c(1, 0.01)</code> .
prior.sigma2	The prior shape and scale in the form of <code>c(shape, scale)</code> for an Inverse-Gamma(shape, scale) prior for sigma2. Defaults to <code>c(1, 0.01)</code> .
MALA	Logical, should the function use MALA (TRUE, default) or simple random walk (FALSE) updates for the random effects. Not applicable if <code>family='gaussian'</code> .
verbose	Logical, should the function update the user on its progress.

**Value**

summary.results	A summary table of the parameters.
samples	A list containing the MCMC samples from the model.
fitted.values	A vector of fitted values for each area.
residuals	A matrix with 3 columns where each column is a type of residual and each row relates to an area. The types are "response" (raw), "pearson", and "deviance".
modelfit	Model fit criteria including the Deviance Information Criterion (DIC) and its corresponding estimated effective number of parameters (p.d), the Log Marginal Predictive Likelihood (LMPL), the Watanabe-Akaike Information Criterion (WAIC) and its corresponding estimated number of effective parameters (p.w), the log-likelihood, and the percentage deviance explained.

accept            The acceptance probabilities for the parameters.  
 localised.structure            NULL, for compatability with the other models.  
 formula            The formula for the covariate and offset part of the model.  
 model            A text string describing the model fit.  
 X            The design matrix of covariates.

### Author(s)

Duncan Lee

### References

Besag, J., J. York, and A. Mollie (1991). Bayesian image restoration with two applications in spatial statistics. *Annals of the Institute of Statistics and Mathematics* 43, 1-59.

### Examples

```
#####
#### Run the model on simulated data on a lattice
#####
#### Load other libraries required
library(MASS)

#### Set up a square lattice region
x.easting <- 1:10
x.northing <- 1:10
Grid <- expand.grid(x.easting, x.northing)
K <- nrow(Grid)

#### set up distance and neighbourhood (W, based on sharing a common border) matrices
distance <- as.matrix(dist(Grid))
W <- array(0, c(K,K))
W[distance==1] <-1

#### Generate the covariates and response data
x1 <- rnorm(K)
x2 <- rnorm(K)
theta <- rnorm(K, sd=0.05)
phi <- mvrnorm(n=1, mu=rep(0,K), Sigma=0.4 * exp(-0.1 * distance))
logit <- x1 + x2 + theta + phi
prob <- exp(logit) / (1 + exp(logit))
trials <- rep(50,K)
Y <- rbinom(n=K, size=trials, prob=prob)

#### Run the BYM model
formula <- Y ~ x1 + x2
## Not run: model <- S.CARbym(formula=formula, family="binomial", trials=trials,
W=W, burnin=20000, n.sample=100000)
## End(Not run)
```

---

S.CARDissimilarity      *Fit a spatial generalised linear mixed model to data, where the random effects have a localised conditional autoregressive prior.*

---

### Description

Fit a spatial generalised linear mixed model to areal unit data, where the response variable can be binomial, Gaussian or Poisson. The linear predictor is modelled by known covariates and a vector of random effects. The latter are modelled by the localised conditional autoregressive prior proposed by Lee and Mitchell (2012), and further details are given in the vignette accompanying this package. Inference is conducted in a Bayesian setting using Markov chain Monte Carlo (MCMC) simulation. Missing (NA) values are allowed in the response, and posterior predictive distributions are created for the missing values for predictive purposes. These are saved in the ‘samples’ argument in the output of the function and are denoted by ‘Y’.

### Usage

```
S.CARDissimilarity(formula, family, data=NULL, trials=NULL, W,
Z, W.binary=TRUE, burnin, n.sample, thin=1, prior.mean.beta=NULL,
prior.var.beta=NULL, prior.nu2=NULL, prior.tau2=NULL, MALA=TRUE, verbose=TRUE)
```

### Arguments

formula	A formula for the covariate part of the model using the syntax of the <code>lm()</code> function. Offsets can be included here using the <code>offset()</code> function. The response can contain missing (NA) values.
family	One of either ‘binomial’, ‘gaussian’ or ‘poisson’, which respectively specify a binomial likelihood model with a logistic link function, a Gaussian likelihood model with an identity link function, or a Poisson likelihood model with a log link function.
data	An optional data.frame containing the variables in the formula.
trials	A vector the same length as the response containing the total number of trials for each area. Only used if family=‘binomial’.
W	A non-negative K by K neighbourhood matrix (where K is the number of spatial units). Typically a binary specification is used, where the jkth element equals one if areas (j, k) are spatially close (e.g. share a common border) and is zero otherwise. For this model only the matrix must be binary.
Z	A list, where each element is a K by K matrix of non-negative dissimilarity metrics.
W.binary	Logical, should the estimated neighbourhood neighbourhood have only binary (0,1) values.
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).

prior.mean.beta	A vector of prior means for the regression parameters beta (Gaussian priors are assumed). Defaults to a vector of zeros.
prior.var.beta	A vector of prior variances for the regression parameters beta (Gaussian priors are assumed). Defaults to a vector with values 100000.
prior.nu2	The prior shape and scale in the form of c(shape, scale) for an Inverse-Gamma(shape, scale) prior for nu2. Defaults to c(1, 0.01) and only used if family='Gaussian'.
prior.tau2	The prior shape and scale in the form of c(shape, scale) for an Inverse-Gamma(shape, scale) prior for tau2. Defaults to c(1, 0.01).
MALA	Logical, should the function use MALA (TRUE, default) or simple random walk (FALSE) updates for the random effects. Not applicable if family='gaussian'.
verbose	Logical, should the function update the user on its progress.

**Value**

summary.results	A summary table of the parameters.
samples	A list containing the MCMC samples from the model.
fitted.values	A vector of fitted values for each area.
residuals	A matrix with 3 columns where each column is a type of residual and each row relates to an area. The types are "response" (raw), "pearson", and "deviance".
modelfit	Model fit criteria including the Deviance Information Criterion (DIC) and its corresponding estimated effective number of parameters (p.d), the Log Marginal Predictive Likelihood (LMPL), the Watanabe-Akaike Information Criterion (WAIC) and its corresponding estimated number of effective parameters (p.w), the log-likelihood, and the percentage deviance explained.
accept	The acceptance probabilities for the parameters.
localised.structure	A list containing two matrices: W.posterior contains posterior medians for each element w_kj of the K by K neighbourhood matrix W; W.border.prob contains posterior probabilities that each w_kj element of the K by K neighbourhood matrix W equals zero. This 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 W elements that correspond to two non-neighbouring areas have NA values.
formula	The formula for the covariate and offset part of the model.
model	A text string describing the model fit.
X	The design matrix of covariates.

**Author(s)**

Duncan Lee

**References**

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

**Examples**

```
#####
#### Run the model on simulated data - localised CAR model
#####
#### Load other libraries required
library(MASS)

#### Set up a square lattice region
x.easting <- 1:10
x.northing <- 1:10
Grid <- expand.grid(x.easting, x.northing)
K <- nrow(Grid)

#### Split the area into two groups between which there will be a boundary.
groups <- rep(1, K)
groups[Grid$Var1>5] <- 2

#### set up distance and neighbourhood (W, based on sharing a common border) matrices
distance <- as.matrix(dist(Grid))
W <- array(0, c(K,K))
W[distance==1] <-1

#### Generate the response data
phi <- mvrnorm(n=1, mu=groups, Sigma=0.2 * exp(-0.1 * distance))
logit <- phi
prob <- exp(logit) / (1 + exp(logit))
trials <- rep(50,K)
Y <- rbinom(n=K, size=trials, prob=prob)

#### Generate a dissimilarity metric
dissimilarity <- cbind(groups) + rnorm(K, sd=0.2)
dissimilarity.matrix <- as.matrix(dist(cbind(dissimilarity, dissimilarity),
method="manhattan", diag=TRUE, upper=TRUE)) * W/2

Z <- list(dissimilarity.matrix=dissimilarity.matrix)

#### Run the localised smoothing model
formula <- Y ~ 1
## Not run: model <- S.CARdissimilarity(formula=formula, family="binomial",
trials=trials, W=W, Z=Z, W.binary=TRUE, burnin=20000, n.sample=100000)
## End(Not run)
```

S.CARleroux

*Fit a spatial generalised linear mixed model to data, where the random effects have a Leroux conditional autoregressive prior.*

**Description**

Fit a spatial generalised linear mixed model to areal unit data, where the response variable can be binomial, Gaussian or Poisson. The linear predictor is modelled by known covariates and a

vector of random effects. The latter are modelled by the conditional autoregressive prior proposed by Leroux et al. (1999), and further details are given in the vignette accompanying this package. Independent random effects can be obtained by setting (`fix.rho=TRUE`, `rho=0`) similar to the old function `S.independent()`, in which case the neighbourhood matrix `W` is not part of the model. In this case enter a fake `W` matrix that is a `K` by `K` matrix of zeros, where `K` is the number of data points. Similarly, the intrinsic CAR model can be obtained by setting (`fix.rho=TRUE`, `rho=1`) similar to the old function `S.CARiar()`. Inference is conducted in a Bayesian setting using Markov chain Monte Carlo (MCMC) simulation. Missing (NA) values are allowed in the response, and posterior predictive distributions are created for the missing values for predictive purposes. These are saved in the 'samples' argument in the output of the function and are denoted by 'Y'.

### Usage

```
S.CARleroux(formula, family, data=NULL, trials=NULL, W, burnin,
n.sample, thin=1, prior.mean.beta=NULL, prior.var.beta=NULL,
prior.nu2=NULL, prior.tau2=NULL, fix.rho=FALSE, rho=NULL, MALA=TRUE, verbose=TRUE)
```

### Arguments

<code>formula</code>	A formula for the covariate part of the model using the syntax of the <code>lm()</code> function. Offsets can be included here using the <code>offset()</code> function. The response can contain missing (NA) values.
<code>family</code>	One of either 'binomial', 'gaussian' or 'poisson', which respectively specify a binomial likelihood model with a logistic link function, a Gaussian likelihood model with an identity link function, or a Poisson likelihood model with a log link function.
<code>data</code>	An optional data.frame containing the variables in the formula.
<code>trials</code>	A vector the same length as the response containing the total number of trials for each area. Only used if <code>family='binomial'</code> .
<code>W</code>	A non-negative <code>K</code> by <code>K</code> neighbourhood matrix (where <code>K</code> is the number of spatial units). Typically a binary specification is used, where the <code>jkth</code> element equals one if areas ( <code>j</code> , <code>k</code> ) are spatially close (e.g. share a common border) and is zero otherwise. The matrix can be non-binary, but each row must contain at least one non-zero entry. Note, if <code>fix.rho=TRUE</code> and <code>rho=0</code> then the <code>W</code> matrix is not actually used by the model so a <code>K</code> by <code>K</code> matrix of zeros could be entered instead if a real <code>W</code> matrix is cannot be obtained.
<code>burnin</code>	The number of MCMC samples to discard as the burn-in period.
<code>n.sample</code>	The number of MCMC samples to generate.
<code>thin</code>	The level of thinning to apply to the MCMC samples to reduce their temporal autocorrelation. Defaults to 1 (no thinning).
<code>prior.mean.beta</code>	A vector of prior means for the regression parameters <code>beta</code> (Gaussian priors are assumed). Defaults to a vector of zeros.
<code>prior.var.beta</code>	A vector of prior variances for the regression parameters <code>beta</code> (Gaussian priors are assumed). Defaults to a vector with values 100000.

prior.nu2	The prior shape and scale in the form of $c(\text{shape}, \text{scale})$ for an Inverse-Gamma(shape, scale) prior for nu2. Defaults to $c(1, 0.01)$ and only used if family='Gaussian'.
prior.tau2	The prior shape and scale in the form of $c(\text{shape}, \text{scale})$ for an Inverse-Gamma(shape, scale) prior for tau2. Defaults to $c(1, 0.01)$ .
fix.rho	Logical, should the spatial dependence parameter rho be fixed rather than being estimated in the model.
rho	The value that the spatial dependence parameter is set to if fix.rho=TRUE. Must be in the interval [0, 1], where 0 corresponds to independence while 1 corresponds to fitting the intrinsic CAR prior.
MALA	Logical, should the function use MALA (TRUE, default) or simple random walk (FALSE) updates for the random effects. Not applicable if family='gaussian'.
verbose	Logical, should the function update the user on its progress.

### Value

summary.results	A summary table of the parameters.
samples	A list containing the MCMC samples from the model.
fitted.values	A vector of fitted values for each area.
residuals	A matrix with 3 columns where each column is a type of residual and each row relates to an area. The types are "response" (raw), "pearson", and "deviance".
modelfit	Model fit criteria including the Deviance Information Criterion (DIC) and its corresponding estimated effective number of parameters (p.d), the Log Marginal Predictive Likelihood (LMPL), the Watanabe-Akaike Information Criterion (WAIC) and its corresponding estimated number of effective parameters (p.w), the log-likelihood, and the percentage deviance explained.
accept	The acceptance probabilities for the parameters.
localised.structure	NULL, for compatability with the other models.
formula	The formula for the covariate and offset part of the model.
model	A text string describing the model fit.
X	The design matrix of covariates.

### Author(s)

Duncan Lee

### References

Leroux, B., X. Lei, and N. Breslow (2000). Estimation of disease rates in small areas: A new mixed model for spatial dependence, Chapter Statistical Models in Epidemiology, the Environment and Clinical Trials, Halloran, M and Berry, D (eds), pp. 135-178. Springer-Verlag, New York.

## Examples

```
#####
#### Run the model on simulated data on a lattice
#####
#### Load other libraries required
library(MASS)

#### Set up a square lattice region
x.easting <- 1:10
x.northing <- 1:10
Grid <- expand.grid(x.easting, x.northing)
K <- nrow(Grid)

#### set up distance and neighbourhood (W, based on sharing a common border) matrices
distance <- as.matrix(dist(Grid))
W <-array(0, c(K,K))
W[distance==1] <-1

#### Generate the covariates and response data
x1 <- rnorm(K)
x2 <- rnorm(K)
theta <- rnorm(K, sd=0.05)
phi <- mvrnorm(n=1, mu=rep(0,K), Sigma=0.4 * exp(-0.1 * distance))
logit <- x1 + x2 + theta + phi
prob <- exp(logit) / (1 + exp(logit))
trials <- rep(50,K)
Y <- rbinom(n=K, size=trials, prob=prob)

#### Run the Leroux model
formula <- Y ~ x1 + x2
## Not run: model <- S.CARleroux(formula=formula, family="binomial",
trials=trials, W=W, burnin=20000, n.sample=100000)
## End(Not run)
```

---

S.CARlocalised

*Fit a spatial generalised linear mixed model to data, where a set of spatially smooth random effects are augmented with a piecewise constant intercept process.*

---

## Description

Fit a spatial generalised linear mixed model to areal unit data, where the response variable can be binomial or Poisson. Note, a Gaussian likelihood is not allowed because of a lack of identifiability among the parameters. The linear predictor is modelled by known covariates, a vector of random effects and a piecewise constant intercept process. The random effects are modelled by an intrinsic CAR prior, while the piecewise constant intercept process was proposed by Lee and Sarran (2015), and allow neighbouring areas to have very different values. Further details are given in the vignette accompanying this package. Inference is conducted in a Bayesian setting using Markov chain Monte Carlo (MCMC) simulation. Missing (NA) values are not allowed in this model.

**Usage**

```
S.CARlocalised(formula, family, data=NULL, G, trials=NULL, W,
burnin, n.sample, thin=1, prior.mean.beta=NULL, prior.var.beta=NULL,
prior.mean.alpha=NULL, prior.var.alpha=NULL, prior.tau2=NULL, prior.delta=NULL,
MALA=TRUE, verbose=TRUE)
```

**Arguments**

formula	A formula for the covariate part of the model using the syntax of the <code>lm()</code> function. Offsets can be included here using the <code>offset()</code> function. The response cannot contain missing (NA) values.
family	One of either 'binomial' or 'poisson', which respectively specify a binomial likelihood model with a logistic link function, or a Poisson likelihood model with a log link function.
data	An optional data.frame containing the variables in the formula.
G	The maximum number of distinct intercept terms (clusters) to allow in the model.
trials	A vector the same length as the response containing the total number of trials for each area. Only used if family='binomial'.
W	A non-negative K by K neighbourhood matrix (where K is the number of spatial units). Typically a binary specification is used, where the jkth element equals one if areas (j, k) are spatially close (e.g. share a common border) and is zero otherwise. The matrix can be non-binary, but each row must contain at least one non-zero entry.
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).
prior.mean.beta	A vector of prior means for the regression parameters beta (Gaussian priors are assumed). Defaults to a vector of zeros.
prior.var.beta	A vector of prior variances for the regression parameters beta (Gaussian priors are assumed). Defaults to a vector with values 1000.
prior.mean.alpha	The prior mean for the regression parameter alpha (Gaussian priors are assumed) multiplying the covariate with within area variation in its values. Defaults to zero.
prior.var.alpha	The prior variance for the regression parameter alpha (Gaussian priors are assumed) multiplying the covariate with within area variation in its values. Defaults to 100000.
prior.tau2	The prior shape and scale in the form of <code>c(shape, scale)</code> for an Inverse-Gamma(shape, scale) prior for tau2. Defaults to <code>c(1, 0.01)</code> .
prior.delta	The prior maximum for the cluster smoothing parameter delta. Defaults to 10.
MALA	Logical, should the function use MALA (TRUE, default) or simple random walk (FALSE) updates for the random effects. Not applicable if family='gaussian'.
verbose	Logical, should the function update the user on its progress.

**Value**

summary.results	A summary table of the parameters.
samples	A list containing the MCMC samples from the model.
fitted.values	A vector of fitted values for each area.
residuals	A matrix with 3 columns where each column is a type of residual and each row relates to an area. The types are "response" (raw), "pearson", and "deviance".
modelfit	Model fit criteria including the Deviance Information Criterion (DIC) and its corresponding estimated effective number of parameters (p.d), the Log Marginal Predictive Likelihood (LMPL), the Watanabe-Akaike Information Criterion (WAIC) and its corresponding estimated number of effective parameters (p.w), the log-likelihood, and the percentage deviance explained.
accept	The acceptance probabilities for the parameters.
localised.structure	A vector giving the posterior median of which cluster (group) each area is in.
formula	The formula for the covariate and offset part of the model.
model	A text string describing the model fit.
X	The design matrix of covariates.

**Author(s)**

Duncan Lee

**References**

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

**Examples**

```
#####
#### Run the model on simulated data - localised CAR model
#####
#### Load other libraries required
library(MASS)

#### Set up a square lattice region
x.easting <- 1:10
x.northing <- 1:10
Grid <- expand.grid(x.easting, x.northing)
K <- nrow(Grid)

#### Split the area into two groups between which there will be a boundary.
groups <- rep(1, K)
groups[Grid$Var1>5] <- 2

#### set up distance and neighbourhood (W, based on sharing a common border) matrices
distance <- as.matrix(dist(Grid))
```

```

W <-array(0, c(K,K))
W[distance==1] <-1

#### Generate the response data
phi <- mvrnorm(n=1, mu=groups, Sigma=0.2 * exp(-0.1 * distance))
logit <- phi
prob <- exp(logit) / (1 + exp(logit))
trials <- rep(50,K)
Y <- rbinom(n=K, size=trials, prob=prob)

#### Run the localised smoothing model
formula <- Y ~ 1
## Not run: model <- S.CARlocalised(formula=formula, family="binomial", trials=trials,
G=2, W=W,burnin=20000, n.sample=100000)
## End(Not run)

```

---

S.CARmultilevel	<i>Fit a spatial generalised linear mixed model to multi-level areal unit data, where the spatial random effects have a Leroux conditional autoregressive prior and there are also individual or small group level random effects.</i>
-----------------	--

---

## Description

Fit a spatial generalised linear mixed model to multi-level areal unit data, where the response variable can be binomial, Gaussian or Poisson. The data are structured with individuals within areal units, and different numbers of individuals are allowed within each areal unit. The linear predictor is modelled by known covariates (either individual or areal level) and two vectors of random effects. The latter include areal level effects modelled by the conditional autoregressive prior proposed by Leroux et al. (1999). Independent random effects can be obtained by setting (fix.rho=TRUE, rho=0). Similarly, the intrinsic CAR model can be obtained by setting (fix.rho=TRUE, rho=1). The second set of random effects are determined by a single effect for each level of a categorical covariate. These random effects are independent with a common variance sigma2. Inference is conducted in a Bayesian setting using Markov chain Monte Carlo (MCMC) simulation. Missing (NA) values are allowed in the response, and posterior predictive distributions are created for the missing values for predictive purposes. These are saved in the 'samples' argument in the output of the function and are denoted by 'Y'.

## Usage

```

S.CARmultilevel(formula, family, data=NULL, trials=NULL, W, ind.area,
ind.re=NULL, burnin, n.sample, thin=1, prior.mean.beta=NULL, prior.var.beta=NULL,
prior.nu2=NULL, prior.tau2=NULL, prior.sigma2=NULL, fix.rho=FALSE, rho=NULL,
verbose=TRUE)

```

**Arguments**

formula	A formula for the covariate part of the model using the syntax of the <code>lm()</code> function. Offsets can be included here using the <code>offset()</code> function. The response can contain missing (NA) values.
family	One of either 'binomial', 'gaussian' or 'poisson', which respectively specify a binomial likelihood model with a logistic link function, a Gaussian likelihood model with an identity link function, or a Poisson likelihood model with a log link function.
data	An optional <code>data.frame</code> containing the variables in the formula.
trials	A vector the same length as the response containing the total number of trials for individual. Only used if <code>family='binomial'</code> .
W	A non-negative K by K neighbourhood matrix (where K is the number of spatial units). Typically a binary specification is used, where the jkth element equals one if areas (j, k) are spatially close (e.g. share a common border) and is zero otherwise. The matrix can be non-binary, but each row must contain at least one non-zero entry. Note, if <code>fix.rho=TRUE</code> and <code>rho=0</code> then the W matrix is not actually used by the model so a K by K matrix of zeros could be entered instead if a real W matrix is cannot be obtained.
ind.area	A vector of integers the same length as the number of data points (individuals) giving which spatial unit (numbered from 1 to K to align with the W matrix) each individual belongs to.
ind.re	A categorical variable where each level will be assigned a different independent random effect. If NULL these random effects are omitted.
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).
prior.mean.beta	A vector of prior means for the regression parameters beta (Gaussian priors are assumed). Defaults to a vector of zeros.
prior.var.beta	A vector of prior variances for the regression parameters beta (Gaussian priors are assumed). Defaults to a vector with values 100000.
prior.nu2	The prior shape and scale in the form of <code>c(shape, scale)</code> for an Inverse-Gamma(shape, scale) prior for nu2. Defaults to <code>c(1, 0.01)</code> and only used if <code>family='Gaussian'</code> .
prior.tau2	The prior shape and scale in the form of <code>c(shape, scale)</code> for an Inverse-Gamma(shape, scale) prior for tau2. Defaults to <code>c(1, 0.01)</code> .
prior.sigma2	The prior shape and scale in the form of <code>c(shape, scale)</code> for an Inverse-Gamma(shape, scale) prior for sigma2. Defaults to <code>c(1, 0.01)</code> .
fix.rho	Logical, should the spatial dependence parameter rho be fixed rather than being estimated in the model.
rho	The value that the spatial dependence parameter is set to if <code>fix.rho=TRUE</code> . Must be in the interval [0, 1], where 0 corresponds to independence while 1 corresponds to fitting the intrinsic CAR prior.
verbose	Logical, should the function update the user on its progress.

**Value**

summary.results	A summary table of the parameters.
samples	A list containing the MCMC samples from the model.
fitted.values	A vector of fitted values for each area.
residuals	A vector of residuals for each area.
modelfit	Model fit criteria including the Deviance Information Criterion (DIC) and its corresponding estimated effective number of parameters (p.d), the Log Marginal Predictive Likelihood (LMPL), the Watanabe-Akaike Information Criterion (WAIC) and its corresponding estimated number of effective parameters (p.w), the log-likelihood, and the percentage deviance explained.
localised.structure	NULL, for compatability with the other models.
formula	The formula for the covariate and offset part of the model.
model	A text string describing the model fit.
X	The design matrix of covariates.

**Author(s)**

Duncan Lee

**Examples**

```
#####
#### Run the model on simulated data on a lattice
#####

#### Set up a square lattice region
x.easting <- 1:10
x.northing <- 1:10
Grid <- expand.grid(x.easting, x.northing)
K <- nrow(Grid)

#### set up distance and neighbourhood (W, based on sharing a common border) matrices
distance <- as.matrix(dist(Grid))
W <- array(0, c(K,K))
W[distance==1] <-1

#### Generate the number of individuals per area and which individuals to which areas
n <- sample(5:30, K, replace=TRUE)
n.total <- sum(n)
ind.area.temp <- rep(1:K, n)
ind.area <- sample(ind.area.temp, n.total, replace=FALSE)

#### Generate a categorical variable
n.levels <- n.total / (0.5*1+0.36*2+0.14*3)
m2 <- round(n.levels * 0.36)
m3 <- round(n.levels * 0.14)
```

```

levels.temp <- c(kronecker(1:m3, rep(1,3)), kronecker((m3+1):(m3+m2), rep(1,2)))
m1 <- n.total-length(levels.temp)
levels.temp2 <- c(levels.temp, (max(levels.temp)+1):(max(levels.temp)+m1))
ind.re <-sample(levels.temp2)
q <- m1+m2+m3

#### Generate the covariates and response data
x1 <- rnorm(n.total)
x2 <- rnorm(n.total)
phi <- mvrnorm(n=1, mu=rep(0,K), Sigma=0.4 * exp(-0.1 * distance))
phi.extend <- phi[ind.area]
zeta <- rnorm(n=q, mean=0, sd=0.1)
zeta.extend <- zeta[ind.re]
logit <- x1 + x2 + phi.extend + zeta.extend
prob <- exp(logit) / (1 + exp(logit))
trials <- rep(50,n.total)
Y <- rbinom(n=n.total, size=trials, prob=prob)

#### Run the model
formula <- Y ~ x1 + x2
## Not run: model <- S.CARmultilevel(formula=formula, family="binomial", ind.area=ind.area,
ind.re=ind.re, trials=trials, W=W, burnin=20000, n.sample=100000)
## End(Not run)

```

---

S.glm

*Fit a spatial generalised linear model to data.*


---

## Description

Fit a spatial generalised linear model to data, where the response variable can be binomial, Gaussian or Poisson. Inference is conducted in a Bayesian setting using Markov chain Monte Carlo (MCMC) simulation. Missing (NA) values are allowed in the response, and posterior predictive distributions are created for the missing values for predictive purposes. These are saved in the 'samples' argument in the output of the function and are denoted by 'Y'.

## Usage

```
S.glm(formula, family, data=NULL, trials=NULL, burnin, n.sample, thin=1,
prior.mean.beta=NULL, prior.var.beta=NULL, prior.nu2=NULL, verbose=TRUE)
```

## Arguments

formula	A formula for the covariate part of the model using the syntax of the <code>lm()</code> function. Offsets can be included here using the <code>offset()</code> function. The response can contain missing (NA) values.
family	One of either 'binomial', 'gaussian' or 'poisson', which respectively specify a binomial likelihood model with a logistic link function, a Gaussian likelihood model with an identity link function, or a Poisson likelihood model with a log link function.

data	An optional data.frame containing the variables in the formula.
trials	A vector the same length as the response containing the total number of trials for data point. Only used if family='binomial'.
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).
prior.mean.beta	A vector of prior means for the regression parameters beta (Gaussian priors are assumed). Defaults to a vector of zeros.
prior.var.beta	A vector of prior variances for the regression parameters beta (Gaussian priors are assumed). Defaults to a vector with values 100000.
prior.nu2	The prior shape and scale in the form of c(shape, scale) for an Inverse-Gamma(shape, scale) prior for nu2. Defaults to c(1, 0.01) and only used if family='Gaussian'.
verbose	Logical, should the function update the user on its progress.

**Value**

summary.results	A summary table of the parameters.
samples	A list containing the MCMC samples from the model.
fitted.values	A vector of fitted values for each area.
residuals	A vector of residuals for each area.
modelfit	Model fit criteria including the Deviance Information Criterion (DIC) and its corresponding estimated effective number of parameters (p.d), the Log Marginal Predictive Likelihood (LMPL), and the Watanabe-Akaike Information Criterion (WAIC) and its corresponding estimated number of effective parameters (p.w).
accept	The acceptance probabilities for the parameters.
localised.structure	NULL, for compatibility with the other models.
formula	The formula for the covariate and offset part of the model.
model	A text string describing the model fit.
X	The design matrix of covariates.

**Author(s)**

Duncan Lee

**Examples**

```
#####
#### Run the model on simulated data on a lattice
#####
#### Set up a square lattice region
x.easting <- 1:10
```

```

x.northing <- 1:10
Grid <- expand.grid(x.easting, x.northing)
K <- nrow(Grid)

#### Generate the covariates and response data
x1 <- rnorm(K)
x2 <- rnorm(K)
logit <- x1 + x2
prob <- exp(logit) / (1 + exp(logit))
trials <- rep(50,K)
Y <- rbinom(n=K, size=trials, prob=prob)

#### Run the model
formula <- Y ~ x1 + x2
## Not run: model <- S.glm(formula=formula, family="binomial", trials=trials,
burnin=20000, n.sample=100000)
## End(Not run)

```

---

summarise.lincomb	<i>Compute the posterior distribution for a linear combination of the covariates from the linear predictor.</i>
-------------------	---

---

## Description

This function takes in a ‘carbeyes’ model object and computes the posterior distribution and posterior quantiles of a linear combination of the covariates from the linear predictor. For example, if a quadratic effect of a covariate on the response was specified, then this function allows you to compute the posterior distribution of the quadratic relationship.

## Usage

```
summarise.lincomb(model, columns=NULL, quantiles=0.5, distribution=FALSE)
```

## Arguments

model	A ‘carbeyes’ model object from fitting one of the models in this package.
columns	A vector of column numbers stating which columns in the design matrix of covariates the posterior distribution should be computed for.
quantiles	The vector of posterior quantiles required.
distribution	A logical value stating whether the entire posterior distribution should be returned or just the specified quantiles.

## Value

quantiles	A 2 dimensional array containing the required posterior quantiles. Each row relates to a data value, and each column to a different requested quantile.
posterior	A 2 dimensional array containing the required posterior distribution. Each column relates to a different data value.

**Author(s)**

Duncan Lee

**Examples**

```
## See the vignette accompanying this package for an example of its use.
```

---

```
summarise.samples      Summarise a matrix of Markov chain Monte Carlo samples.
```

---

**Description**

This function takes in a matrix of Markov chain Monte Carlo (MCMC) samples from a ‘carbays’ model object, such as a set of parameters or fitted values, and calculates posterior quantiles and exceedence probabilities. The latter are probabilities of the form  $P(\text{quantity} > \text{cldata})$ , where  $c$  is a threshold chosen by the user.

**Usage**

```
summarise.samples(samples, columns=NULL, quantiles=0.5, exceedences=NULL)
```

**Arguments**

samples	A matrix of MCMC samples obtained from a ‘carbays’ model object.
columns	A vector of column numbers stating which columns in the matrix of MCMC samples summaries are required for. Defaults to all columns.
quantiles	The vector of posterior quantiles required.
exceedences	The vector of threshold levels, $c$ , that exceedence probabilities are required for.

**Value**

quantiles	A 2 dimensional array containing the required posterior quantiles. Each row relates to a parameter and each column to a different requested quantile.
exceedences	A 2 dimensional array containing the required exceedence probabilities. Each row relates to a parameter, and each column to a different requested exceedence probability.

**Author(s)**

Duncan Lee

**Examples**

```
## See the vignette accompanying this package for an example of its use.
```

# Index

CARBayes (CARBayes-package), [2](#)  
CARBayes-package, [2](#)  
coef.CARBayes, [4](#)  
combine.data.shapefile, [4](#)  
  
fitted.CARBayes, [5](#)  
  
highlight.borders, [6](#)  
  
logLik.CARBayes, [7](#)  
  
model.matrix.CARBayes, [7](#)  
MVS.CARleroux, [8](#)  
  
print.CARBayes, [11](#)  
  
residuals.CARBayes, [12](#)  
  
S.CARbym, [12](#)  
S.CARDissimilarity, [15](#)  
S.CARleroux, [17](#)  
S.CARlocalised, [20](#)  
S.CARmultilevel, [23](#)  
S.glm, [26](#)  
summarise.lincomb, [28](#)  
summarise.samples, [29](#)