

Package ‘CARBayes’

June 30, 2015

Type Package

Title Spatial Generalised Linear Mixed Models for Areal Unit Data

Version 4.2

Date 2015-06-30

Author Duncan Lee

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

Description Implements a class of 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. The spatial autocorrelation is modelled by a set of random effects, which are assigned a conditional autoregressive (CAR) prior distribution. A number of different CAR priors are available for the random effects, and 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 is supported by the Engineering and Physical Science Research Council (EPSRC) grant EP/J017442/1.

License GPL (≥ 2)

Depends MASS, R ($\geq 2.10.0$), Rcpp ($\geq 0.10.6$)

Imports CARBayesdata, coda, sp, spam, spdep, stats, truncdist, utils

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

LinkingTo Rcpp

LazyLoad yes

NeedsCompilation yes

Repository CRAN

Date/Publication 2015-06-30 18:13:48

R topics documented:

CARBayes-package	2
combine.data.shapefile	3
highlight.borders	4

print.carbayes	5
S.CARbym	5
S.CARdissimilarity	8
S.CARiar	10
S.CARleroux	13
S.CARlocalised	15
S.independent	18
summarise.lincomb	20
summarise.samples	21

Index	23
--------------	-----------

CARBayes-package	<i>Spatial Generalised Linear Mixed Models for Areal Unit Data</i>
------------------	--------------------------------------------------------------------

Description

Implements a class of 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. The spatial autocorrelation is modelled by a set of random effects, which are assigned a conditional autoregressive (CAR) prior distribution. A number of different CAR priors are available for the random effects, and 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 is supported by the Engineering and Physical Science Research Council (EPSRC) grant EP/J017442/1.

Details

Package: CARBayes
 Type: Package
 Version: 4.2
 Date: 2015-06-30
 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.

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*, to appear.

Leroux, B., Lei, X and Breslow, N (1999). 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

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

combine.data.shapefile

Combines a data frame with a shapefile to create a SpatialPolygons-DataFrame object.

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 ‘`spplot()`’ 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 <code>K</code> 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 <code>K</code> areas that the data relate to.
dbf	The .dbf part of the shapefile containing a lookup table whose first column should include the <code>K</code> 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.
```

<code>highlight.borders</code>	<i>Creates a <code>SpatialPoints</code> 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

<code>border.locations</code>	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.
<code>spdata</code>	The <code>'SpatialPolygonsDataFrame'</code> 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.
```

```
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

```
S.CARbym          Fit a spatial generalised linear mixed model to data, where the random effects have a BYM conditional autoregressive prior.
```

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.

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, 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.
<code>family</code>	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.
<code>data</code>	An optional <code>data.frame</code> 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 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 burnin 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.
<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 1000.
<code>prior.tau2</code>	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(0.001, 0.001)</code> .
<code>prior.sigma2</code>	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(0.001, 0.001)</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.
<code>residuals</code>	A vector of residuals for each area.
<code>modelfit</code>	Model fit criteria including the Deviance Information Criterion (DIC), the effective number of parameters in the model (p.d), and the Log Marginal Predictive Likelihood (LMPL).
<code>accept</code>	The acceptance probabilities for the parameters.
<code>localised.structure</code>	NULL, for compatibility with the other models.
<code>formula</code>	The formula for the covariate and offset part of the model.
<code>model</code>	A text string describing the model fit.
<code>X</code>	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
#####

#### 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 <- array(0, c(K,K))
W <- array(0, c(K,K))
  for(i in 1:K)
  {
  for(j in 1:K)
  {
  temp <- (Grid[i,1] - Grid[j,1])^2 + (Grid[i,2] - Grid[j,2])^2
  distance[i,j] <- sqrt(temp)
  if(temp==1) W[i,j] <- 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.

Usage

```
S.CARdissimilarity(formula, family, data=NULL, trials=NULL, W,
Z, burnin, n.sample, thin=1, prior.mean.beta=NULL,
prior.var.beta=NULL, prior.nu2=NULL, prior.tau2=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.
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 each area. Only used if <code>family='binomial'</code> .
W	A K by K neighbourhood matrix (where K is the number of spatial units). Typically a binary specification is used, where the <code>jkth</code> 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.
burnin	The number of McMC samples to discard as the burnin 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.
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.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 $c(0.001, 0.001)$ 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 $c(0.001, 0.001)$.
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), the effective number of parameters in the model (p.d), and the Log Marginal Predictive Likelihood (LMPL).
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. In both cases elements which 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
#####

#### 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 <- array(0, c(K,K))
W <- array(0, c(K,K))
for(i in 1:K)
{
for(j in 1:K)
{
temp <- (Grid[i,1] - Grid[j,1])^2 + (Grid[i,2] - Grid[j,2])^2
distance[i,j] <- sqrt(temp)
if(temp==1) W[i,j] <- 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, burnin=20000, n.sample=100000)
## End(Not run)

```

S.CARiar

Fit a spatial generalised linear mixed model to data, where the random effects have an intrinsic 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 intrinsic 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.

Usage

```
S.CARiar(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, 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.
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 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 burnin 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.
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.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(0.001, 0.001)</code> and only used if family='Gaussian'.
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(0.001, 0.001)</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 vector of residuals for each area.
modelfit	Model fit criteria including the Deviance Information Criterion (DIC), the effective number of parameters in the model (p.d), and the Log Marginal Predictive Likelihood (LMPL).
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
#####

#### 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 <-array(0, c(K,K))
W <-array(0, c(K,K))
for(i in 1:K)
{
for(j in 1:K)
{
temp <- (Grid[i,1] - Grid[j,1])^2 + (Grid[i,2] - Grid[j,2])^2
distance[i,j] <- sqrt(temp)
if(temp==1) W[i,j] <- 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 IAR model
formula <- Y ~ x1 + x2
## Not run: model <- S.CARlar(formula=formula, family="binomial", trials=trials,
W=W, 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. Inference is conducted in a Bayesian setting using Markov chain Monte Carlo (MCMC) simulation.

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, 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.
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 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 burnin 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.
<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 1000.
<code>prior.nu2</code>	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(0.001, 0.001)</code> and only used if family='Gaussian'.
<code>prior.tau2</code>	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(0.001, 0.001)</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.
<code>residuals</code>	A vector of residuals for each area.
<code>modelfit</code>	Model fit criteria including the Deviance Information Criterion (DIC), the effective number of parameters in the model (p.d), and the Log Marginal Predictive Likelihood (LMPL).
<code>accept</code>	The acceptance probabilities for the parameters.
<code>localised.structure</code>	NULL, for compatability with the other models.
<code>formula</code>	The formula for the covariate and offset part of the model.
<code>model</code>	A text string describing the model fit.
<code>X</code>	The design matrix of covariates.

Author(s)

Duncan Lee

References

Leroux, B., X. Lei, and N. Breslow (1999). 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
#####

#### 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 <- array(0, c(K,K))
W <- array(0, c(K,K))
for(i in 1:K)
{
  for(j in 1:K)
  {
    temp <- (Grid[i,1] - Grid[j,1])^2 + (Grid[i,2] - Grid[j,2])^2
    distance[i,j] <- sqrt(temp)
    if(temp==1) W[i,j] <- 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 the random effects have conditional autoregressive prior and are augmented with a piecewise constant intercept term.

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.

Usage

```
S.CARlocalised(formula, exposure=NULL, 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,
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.
exposure	This optional argument allows a single covariate to have variation in its values within an areal unit if <code>family='poisson'</code> . The argument is a list object, and each element of the list corresponds to a single areal unit. Each element in the list is a 2 column matrix, where the first column is the vector of values of the covariate while the second column is a vector of weights for each value.
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 <code>data.frame</code> 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 <code>family='binomial'</code> .
W	A 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 burnin 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.
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 1000.
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(0.001, 0.001).
prior.delta	The prior maximum for the cluster smoothing parameter delta. Defaults to 10.
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), the effective number of parameters in the model (p.d), and the Log Marginal Predictive Likelihood (LMPL).
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*, to appear.

Examples

```
#####
#### Run the model on simulated data - localised CAR model
#####

#### 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 <-array(0, c(K,K))
W <-array(0, c(K,K))
  for(i in 1:K)
  {
for(j in 1:K)
{
temp <- (Grid[i,1] - Grid[j,1])^2 + (Grid[i,2] - Grid[j,2])^2
distance[i,j] <- sqrt(temp)
if(temp==1) W[i,j] <- 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", G=2,
W=W,burnin=20000, n.sample=100000)
## End(Not run)
```

S.independent

Fit a spatial generalised linear mixed model to data, where the random effects are independent.

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 as independent and identically distributed, 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.

Usage

```
S.independent(formula, family, data=NULL, trials=NULL, burnin, n.sample, thin=1,
prior.mean.beta=NULL, prior.var.beta=NULL, prior.sigma2=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.
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 <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 each area. Only used if <code>family='binomial'</code> .
burnin	The number of MCMC samples to discard as the burnin 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.
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.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(0.001, 0.001)</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 vector of residuals for each area.
modelfit	Model fit criteria including the Deviance Information Criterion (DIC), the effective number of parameters in the model (p.d), and the Log Marginal Predictive Likelihood (LMPL).
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)
theta <- rnorm(K, sd=0.05)
logit <- x1 + x2 + theta
prob <- exp(logit) / (1 + exp(logit))
trials <- rep(50,K)
Y <- rbinom(n=K, size=trials, prob=prob)

#### Run the independent model
formula <- Y ~ x1 + x2
## Not run: model <- S.independent(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	<i>Summarise a matrix of Markov chain Monte Carlo samples.</i>
-------------------	----------------------------------------------------------------

Description

This function takes in a matrix of Markov chain Monte Carlo (MCMC) samples from a ‘carbeyes’ 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 ‘carbeyes’ 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](#)

combine.data.shapefile, [3](#)

highlight.borders, [4](#)

print.carbayes, [5](#)

S.CARbym, [5](#)

S.CARdissimilarity, [8](#)

S.CARiar, [10](#)

S.CARleroux, [13](#)

S.CARlocalised, [15](#)

S.independent, [18](#)

summarise.lincomb, [20](#)

summarise.samples, [21](#)