

Package ‘geoBayes’

May 13, 2018

Type Package

Title Analysis of Geostatistical Data using Bayes and Empirical Bayes Methods

Description Functions to fit geostatistical data. The data can be continuous, binary or count data and the models implemented are flexible. Conjugate priors are assumed on some parameters while inference on the other parameters can be done through a full Bayesian analysis of by empirical Bayes methods.

Version 0.5.1

Date 2018-05-10

Author Evangelos Evangelou <e.evangelou@maths.bath.ac.uk>, Vivekananda Roy <vro@iastate.edu>

Maintainer Evangelos Evangelou <e.evangelou@maths.bath.ac.uk>

Imports stats, coda, sp

Depends R (>= 3.0.0)

Suggests geoR

LazyData true

License GPL (>= 2)

Classification/ACM 86A32, 62M30, 62F15

RoxygenNote 6.0.1

NeedsCompilation yes

Repository CRAN

Date/Publication 2018-05-13 18:02:31 UTC

R topics documented:

| | |
|--------------------|---|
| bf1skel | 2 |
| bf2new | 4 |
| bf2optim | 6 |
| bf2se | 7 |

| | |
|--------------------------------|----|
| bfse | 8 |
| comparebinlinks | 9 |
| geoBayes | 10 |
| geoBayes_correlation | 11 |
| geoBayes_models | 12 |
| likaprxn | 12 |
| likoptim | 13 |
| linkfcn | 15 |
| mcmcmake | 16 |
| mcsghmm | 18 |
| mcstrga | 21 |
| mkpredgrid2d | 25 |
| plotbf2 | 26 |
| revlogreg | 28 |
| rhizoctonia | 28 |
| rsglmm | 29 |
| sploglik | 31 |
| stackdata | 32 |
| subset.geomcmc | 33 |
| transfsample | 33 |

| | |
|--------------|-----------|
| Index | 35 |
|--------------|-----------|

| | |
|---------|--|
| bf1skel | <i>Computation of Bayes factors at the skeleton points</i> |
|---------|--|

Description

Function to compute the Bayes factors from MCMC samples.

Usage

```
bf1skel(runs, bftype1 = 0.8, method = c("RL", "MW"), reference = 1,
        transf = c("no", "mu", "wo"))
```

Arguments

| | |
|-----------|---|
| runs | A list with outputs from the function <code>mcsghmm</code> or <code>mcstrga</code> . |
| bftype1 | A scalar or vector of the same length as runs with all integer values or all values in (0, 1]. How many samples (or what proportion of the sample) to use for estimating the Bayes factors at the first stage. The remaining sample will be used for estimating the Bayes factors in the second stage. Setting it to 1 will perform only the first stage. |
| method | Which method to use to calculate the Bayes factors: Reverse logistic or Meng-Wong. |
| reference | Which model goes in the denominator. |

`transf` Whether to use a transformed sample for the computations. If "no" or FALSE, it doesn't. If "mu" or TRUE, it uses the samples for the mean. If "wo" it uses an alternative transformation. The latter can be used only for the families indicated by `.geoBayes_models$haswo`.

Details

Computes the Bayes factors using method with respect to reference.

Value

A list with components

- `logbf` A vector containing logarithm of the Bayes factors.
- `logLik1` `logLik2` Matrices with the values of the log-likelihood computed from the samples for each model at the first and second stages.
- `isweights` A vector with the importance sampling weights for computing the Bayes factors at new points that will be used at the second stage. Used internally in `bf2new` and `bf2optim`.
- `controlvar` A matrix with the control variates computed at the samples that will be used in the second stage.
- `sample2` The MCMC sample for μ or z that will be used in the second stage. Used internally in `bf2new` and `bf2optim`.
- `N1`, `N2` Vectors containing the sample sizes used in the first and second stages.
- `distmat` Matrix of distances between locations.
- `betm0`, `betQ0`, `ssqdf`, `ssqsc`, `tsqdf`, `tsqsc`, `dispersion`, `response`, `weights`, `modelmatrix`, `locations`, `family`, `corrfcn`, `transf` Model parameters used internally in `bf2new` and `bf2optim`.
- `pnts` A list containing the skeleton points. Used internally in `bf2new` and `bf2optim`.

References

- Geyer, C. J. (1994). Estimating normalizing constants and reweighting mixtures. Technical report, University of Minnesota.
- Meng, X. L., & Wong, W. H. (1996). Simulating ratios of normalizing constants via a simple identity: A theoretical exploration. *Statistica Sinica*, 6, 831-860.
- Roy, V., Evangelou, E., and Zhu, Z. (2015). Efficient estimation and prediction for the Bayesian spatial generalized linear mixed model with flexible link functions. *Biometrics*, 72(1), 289-298.

Examples

```
## Not run:
data(rhizoctonia)
### Define the model
corrfunc <- "spherical"
kappa <- 0
ssqdf <- 1
ssqsc <- 1
```

```

betm0 <- 0
betQ0 <- .01
linkp <- "probit"
### Skeleton points
phillist <- c(100, 140, 180)
omglist <- c(.5, 1)
parlist <- expand.grid(phi=phillist, linkp=linkp, omg=omglist, kappa = kappa)
### MCMC sizes
Nout <- 100
Nthin <- 1
Nbi <- 0
### Take MCMC samples
runs <- list()
for (i in 1:NROW(parlist)) {
  runs[[i]] <- mcsglmm(Infected ~ 1, 'binomial', rhizoctonia, weights = Total,
    atsample = ~ Xcoord + Ycoord,
    Nout = Nout, Nthin = Nthin, Nbi = Nbi,
    betm0 = betm0, betQ0 = betQ0,
    ssqdf = ssqdf, ssqsc = ssqsc,
    phistart = parlist$phi[i], omgstart = parlist$omg[i],
    linkp = parlist$linkp[i], kappa = parlist$kappa[i],
    corrfcn = corrf, phisc = 0, omgsc = 0)
}
bf <- bf1skel(runs)
bf$logbf

## End(Not run)

```

bf2new

Compute the Bayes factors at new points

Description

Compute the Bayes factors.

Usage

```
bf2new(bf1obj, linkp, phi, omg, kappa, useCV = TRUE)
```

Arguments

| | |
|------------------------|---|
| bf1obj | Output from the function bf1skel which contains the Bayes factors and importance sampling weights. |
| linkp, phi, omg, kappa | Optional scalar or vector or NULL. If scalar or vector, the Bayes factors are calculated at those values with respect to the reference model used in bf1skel . If missing or NULL then the unique values from the MCMC chains that were inputted in bf1skel will be used. |
| useCV | Whether to use control variates for finer corrections. |

Details

Computes the Bayes factors using the importance weights at the new points. The new points are taken from the grid derived by expanding the parameter values inputted. The arguments linkp phi omg kappa correspond to the link function, spatial range, relative nugget, and correlation function parameters respectively.

Value

An array of size $\text{length}(\text{linkp}) * \text{length}(\text{phi}) * \text{length}(\text{omg}) * \text{length}(\text{kappa})$ containing the Bayes factors for each combination of the parameters.

References

Doss, H. (2010). Estimation of large families of Bayes factors from Markov chain output. *Statistica Sinica*, 20(2), 537.

Roy, V., Evangelou, E., and Zhu, Z. (2015). Efficient estimation and prediction for the Bayesian spatial generalized linear mixed model with flexible link functions. *Biometrics*, 72(1), 289-298.

Examples

```
## Not run:
data(rhizoctonia)
### Define the model
corrf <- "spherical"
kappa <- 0
ssqdf <- 1
ssqsc <- 1
betm0 <- 0
betQ0 <- .01
linkp <- "probit"
### Skeleton points
philist <- c(100, 140, 180)
omglist <- c(.5, 1)
parlist <- expand.grid(phi=philist, linkp=linkp, omg=omglist, kappa = kappa)
### MCMC sizes
Nout <- 100
Nthin <- 1
Nbi <- 0
### Take MCMC samples
runs <- list()
for (i in 1:NROW(parlist)) {
  runs[[i]] <- mcsglmm(Infected ~ 1, 'binomial', rhizoctonia, weights = Total,
    atsample = ~ Xcoord + Ycoord,
    Nout = Nout, Nthin = Nthin, Nbi = Nbi,
    betm0 = betm0, betQ0 = betQ0,
    ssqdf = ssqdf, ssqsc = ssqsc,
    phistart = parlist$phi[i], omgstart = parlist$omg[i],
    linkp = parlist$linkp[i], kappa = parlist$kappa[i],
    corrfcn = corrf, phisc = 0, omgsc = 0)
}
bf <- bf1skel(runs)
```

```
bfall <- bf2new(bf, phi = seq(100, 200, 10), omg = seq(0, 2, .2))
plotbf2(bfall, c("phi", "omg"))

## End(Not run)
```

bf2optim

Empirical Bayes estimator

Description

Estimation by empirical Bayes.

Usage

```
bf2optim(bf1obj, paroptim, useCV = TRUE, control = list())
```

Arguments

| | |
|----------|---|
| bf1obj | Output from the function bf1skel which contains the Bayes factors and importance sampling weights. |
| paroptim | A named list with the components "linkp", "phi", "omg", "kappa". Each component must be numeric with length 1, 2, or 3 with elements in increasing order but for the binomial family linkp is also allowed to be the character "logit" and "probit". If the component's length is 1, then the corresponding parameter is considered to be fixed at that value. If 2, then the two numbers denote the lower and upper bounds for the optimisation of that parameter (infinities are allowed). If 3, these correspond to lower bound, starting value, upper bound for the estimation of that parameter. |
| useCV | Whether to use control variates for finer corrections. |
| control | A list of control parameters for the optimisation. See optim . |

Details

This function is a wrap around [bf2new](#) using the "L-BFGS-B" method of the function [optim](#) to estimate the parameters.

Value

The output from the function [optim](#). The "value" element is the log-Bayes factor, not the negative log-Bayes factor.

Examples

```

## Not run:
data(rhizoctonia)
### Define the model
corrfunc <- "spherical"
kappa <- 0
ssqdf <- 1
ssqsc <- 1
betm0 <- 0
betQ0 <- .01
linkp <- "probit"
### Skeleton points
phillist <- c(100, 140, 180)
omglist <- c(.5, 1)
parlist <- expand.grid(phi=phillist, linkp=linkp, omg=omglist, kappa = kappa)
### MCMC sizes
Nout <- 100
Nthin <- 1
Nbi <- 0
### Take MCMC samples
runs <- list()
for (i in 1:NROW(parlist)) {
  runs[[i]] <- mcsglmm(Infected ~ 1, 'binomial', rhizoctonia, weights = Total,
    atsample = ~ Xcoord + Ycoord,
    Nout = Nout, Nthin = Nthin, Nbi = Nbi,
    betm0 = betm0, betQ0 = betQ0,
    ssqdf = ssqdf, ssqsc = ssqsc,
    phistart = parlist$phi[i], omgstart = parlist$omg[i],
    linkp = parlist$linkp[i], kappa = parlist$kappa[i],
    corrfcn = corrfunc, phisc = 0, omgsc = 0)
}
bf <- bf1skel(runs)
est <- bf2optim(bf, list(linkp = linkp, phi = c(100, 200), omg = c(0, 2)))
est

## End(Not run)

```

bf2se

SE for BF

Description

Standard errors for BF estimates

Usage

```
bf2se(mcrun, transf = c("no", "mu", "wo"))
```

Arguments

| | |
|--------|--|
| mcrun | The output from the function <code>mcsglm</code> . |
| transf | The type of transformation to use. |

Details

Using the formula from Casella.

Value

The precision matrix

 bfse

Computation of standard errors for Bayes factors

Description

Compute the standard errors for the Bayes factors estimates.

Usage

```
bfse(pargrid, runs, bftype1 = 0.8, nbatch1 = 0.5, nbatch2 = 0.5,
     method = c("RL", "MW"), bvmethod = c("Standard", "TukeyHanning",
     "Bartlett"), reference = 1, transf = c("no", "mu", "wo"))
```

Arguments

| | |
|---------|---|
| pargrid | A data frame with components "linkp", "phi", "omg", "kappa". Each row gives a combination of the parameters to compute the new standard errors. |
| runs | A list with outputs from the function <code>mcsglm</code> or <code>mcstrga</code> . |
| bftype1 | A scalar or vector of the same length as runs with all integer values or all values in (0, 1]. How many samples (or what proportion of the sample) to use for estimating the Bayes factors at the first stage. The remaining sample will be used for estimating the standard errors in the second stage. Setting it to 1 will perform only the first stage. |
| nbatch1 | A scalar or vector of the same length as runs. All values must be integers or less than 1. This is used for calculating how many batches to split each of the sample in runs for the calculation of the Bayes factors standard errors for the parameters corresponding to runs. |
| nbatch2 | A scalar or vector of the same length as runs. All values must be integers or less than 1. This is used for calculating how many batches to split each of the sample in runs for the calculation of the Bayes factors standard errors for the parameters corresponding to pargrid. |
| method | Which method to use to calculate the Bayes factors: Reverse logistic or Meng-Wong. |

| | |
|-----------|--|
| bvmethod | Which method to use for the calculation of the batch variance. The standard method splits to disjoint batches. The second and third method use the spectral variance method with different lag windows. |
| reference | Which model goes in the denominator. |
| transf | Whether to use a transformed sample for the computations. If "no" or FALSE, it doesn't. If "mu" or TRUE, it uses the samples for the mean. If "wo" it uses an alternative transformation. The latter can be used only for the families indicated by <code>.geoBayes_models\$haswo</code> . |

Details

Uses the batch means method to compute the standard errors for Bayes factors.

Value

A list with components

- `pargrid` The inputted pargrid augmented with the computed standard errors.
- `bfEstimate` The estimates of the Bayes factors
- `bfSigma` The covariance matrix for the Bayes factors estimates.

References

Roy, V., Tan, A. and Flegal, J. (2018). Estimating standard errors for importance sampling estimators with multiple Markov chains, *Statistica Sinica*, 28 1079-1101.

| | |
|-----------------|---|
| comparebinlinks | <i>Compare two link functions for the binomial distribution</i> |
|-----------------|---|

Description

Compare two link functions for the binomial distribution.

Usage

```
comparebinlinks(d1, d2, a = -8, b = 8, n = 2001, prob1 = NULL,
  plot = TRUE)
```

Arguments

| | |
|----|---|
| d1 | Either a positive number or one of "logit" or "probit" defining the link function. If positive number, this corresponds to the <code>robit(d1)</code> link. |
| d2 | Same as d1. |
| a | Beginning of range of points to evaluate (and plot) the two link functions. |
| b | End of range of points to evaluate (and plot) the two link functions. |
| n | Number of points for evaluating (and plot) the two link functions. |

| | |
|-------|---|
| prob1 | Binomial probabilities corresponding to the first link function. |
| plot | Whether the two link functions should be plotted. If so, two plots are created: the left plot shows two curves corresponding to the two link functions (black is the first) and the right plot shows their difference (first - second). |

Details

Finds the closest (after suitable scaling) of the link function d2 to d1. If prob1 is provided, then this corresponds to probabilities under link d1 and the probabilities are transformed to d2. Also creates a plot of the two link functions.

Value

A list with the following elements

- scale The optimal scaling c such that $\max(\text{abs}(\text{link}(x/c, d1) - \text{link}(x, d2)))$ is minimised for $x = \text{seq}(a, b, \text{length.out}=n)$
- maxdiff The maximum difference between the two links.
- prob2 The corresponding probabilities from prob1 to the second link.

Examples

```
## Not run:
comparebinlinks("logit", 7) # The robit(7) approximates logit
comparebinlinks("probit", 1, prob1 = c(.5, .6, .7, .8, .9))

## End(Not run)
```

geoBayes

The geoBayes package

Description

Analysis of geostatistical data using Bayes and Empirical Bayes methods.

Details

This package provides functions to fit geostatistical data. The data can be continuous, binary or count data and the models implemented are flexible. Conjugate priors are assumed on some parameters while inference on the other parameters can be done through a full Bayesian analysis of by empirical Bayes methods.

Some demonstration examples are provided. Type `demo(package = "geoBayes")` to examine them.

Author(s)

Evangelos Evangelou <e.evangelou@maths.bath.ac.uk> and Vivekananda Roy <vro@iastate.edu>

References

Roy, V., Evangelou, E. and Zhu, Z. (2014). Empirical Bayes methods for the transformed Gaussian random fields model with additive measurement errors. In Upadhyay, S. K., Singh, U., Dey, D. K., and Loganathan, A., editors, *Current Trends in Bayesian Methodology with Applications*, Boca Raton, FL, USA, CRC Press.

Roy, V., Evangelou, E., and Zhu, Z. (2015). Efficient estimation and prediction for the Bayesian spatial generalized linear mixed model with flexible link functions. *Biometrics*, 72(1), 289-298.

See Also

geoR, geoRglm

Examples

```
## Not run:  
demo(package = "geoBayes")  
demo(rhizoctonia3, package = "geoBayes")  
  
## End(Not run)
```

geoBayes_correlation *Spatial correlation used in the geoBayes package*

Description

This hidden variable contains a choice of correlation functions that can be fit with this package. The function can be chosen in the `corrfcn` input of the relevant function. This variable cannot be overwritten.

Usage

```
.geoBayes_corrfcn
```

Format

An object of class `data.frame` with 3 rows and 4 columns.

| | |
|-----------------|--|
| geoBayes_models | <i>Models used in the geoBayes package</i> |
|-----------------|--|

Description

This hidden variable contains a choice of models that can be fit with this package. The model can be chosen in the family input of the relevant function. This variable cannot be overwritten.

Usage

```
.geoBayes_models
```

Format

An object of class `data.frame` with 11 rows and 7 columns.

| | |
|----------|-------------------------------------|
| likaprxn | <i>Log-likelihood approximation</i> |
|----------|-------------------------------------|

Description

Log-likelihood approximation

Usage

```
likaprxn(linkp, phi, omg, kappa, formula, family = "gaussian", data, weights,
  subset, atsample, corrfcn = "matern", np, betm0, betQ0, ssqdf, ssqsc, tsqdf,
  tsqsc, dispersion = 1, longlat = FALSE)
```

Arguments

| | |
|---------|---|
| linkp | Parameter of the link function. For binomial, a positive number for the degrees of freedom of the robit family or "logit" or "probit". For the other families any number for the exponent of the Box-Cox transformation. Input can be a scalar or a vector. |
| phi | Spatial range parameter. Input can be a scalar or a vector. |
| omg | Relative nugget parameter. Input can be a scalar or a vector. |
| kappa | Spatial smoothness parameter. Input can be a scalar or a vector. |
| formula | A representation of the model in the form <code>response ~ terms</code> . |
| family | The distribution of the response. Can be one of the options in .geoBayes_models or "transformed.gaussian". |
| data | An optional data frame containing the variables in the model. |
| weights | An optional vector of weights. Number of replicated samples for Gaussian and gamma, number of trials for binomial, time length for Poisson. |

| | |
|------------|---|
| subset | An optional vector specifying a subset of observations to be used in the fitting process. |
| atsample | A formula in the form $\sim x_1 + x_2 + \dots + x_d$ with the coordinates of the sampled locations. |
| corrfcn | Spatial correlation function. Can be one of the choices in <code>.geoBayes_corrfcn</code> . |
| np | The number of integration points for the spatial variance parameter σ^2 . The total number of points will be $2 * np + 1$. |
| betm0 | Prior mean for beta (a vector or scalar). |
| betQ0 | Prior standardised precision (inverse variance) matrix. Can be a scalar, vector or matrix. The first two imply a diagonal with those elements. Set this to 0 to indicate a flat improper prior. |
| ssqdf | Degrees of freedom for the scaled inverse chi-square prior for the partial sill parameter. |
| ssqsc | Scale for the scaled inverse chi-square prior for the partial sill parameter. |
| tsqdf | Degrees of freedom for the scaled inverse chi-square prior for the measurement error parameter. |
| tsqsc | Scale for the scaled inverse chi-square prior for the measurement error parameter. |
| dispersion | The fixed dispersion parameter. |
| longlat | How to compute the distance between locations. If FALSE, Euclidean distance, if TRUE Great Circle distance. See <code>spDists</code> . |

Details

Computes and approximation to the log-likelihood for the given parameters.

Value

A vector of the same length as the parameters containing the log-likelihood values.

| | |
|----------|------------------------------------|
| likoptim | <i>Log-likelihood maximisation</i> |
|----------|------------------------------------|

Description

Log-likelihood maximisation

Usage

```
likoptim(paroptim, formula, family = "gaussian", data, weights, subset,
  atsample, corrfcn = "matern", np, betm0, betQ0, ssqdf, ssqsc,
  dispersion = 1, longlat = FALSE, control = list())
```

Arguments

| | |
|------------|---|
| paroptim | A named list with the components "linkp", "phi", "omg", "kappa". Each component must be numeric with length 1, 2, or 3 with elements in increasing order but for the binomial family linkp is also allowed to be the character "logit" and "probit". If the component's length is 1, then the corresponding parameter is considered to be fixed at that value. If 2, then the two numbers denote the lower and upper bounds for the optimisation of that parameter (infinities are allowed). If 3, these correspond to lower bound, starting value, upper bound for the estimation of that parameter. |
| formula | A representation of the model in the form <code>response ~ terms</code> . |
| family | The distribution of the response. |
| data | An optional data frame containing the variables in the model. |
| weights | An optional vector of weights. Number of replicated samples for Gaussian and gamma, number of trials for binomial, time length for Poisson. |
| subset | An optional vector specifying a subset of observations to be used in the fitting process. |
| atsample | A formula in the form <code>~ x1 + x2 + ... + xd</code> with the coordinates of the sampled locations. |
| corrfcn | Spatial correlation function. See geoBayes_correlation for details. |
| np | The number of integration points for the spatial variance parameter σ^2 . The total number of points will be $2 * np + 1$. |
| betm0 | Prior mean for beta (a vector or scalar). |
| betQ0 | Prior standardised precision (inverse variance) matrix. Can be a scalar, vector or matrix. The first two imply a diagonal with those elements. Set this to 0 to indicate a flat improper prior. |
| ssqdf | Degrees of freedom for the scaled inverse chi-square prior for the partial sill parameter. |
| ssqsc | Scale for the scaled inverse chi-square prior for the partial sill parameter. |
| dispersion | The fixed dispersion parameter. |
| longlat | How to compute the distance between locations. If FALSE, Euclidean distance, if TRUE Great Circle distance. See spDists . |
| control | A list of control parameters for the optimisation. See optim . |

Details

Uses the "L-BFGS-B" method of the function [optim](#) to maximise the log-likelihood for the parameters linkp, phi, omg, kappa.

Value

The output from the function [optim](#). The "value" element is the log-likelihood, not the negative log-likelihood.

| | |
|---------|---|
| linkfcn | <i>Calculate the link function for exponential families</i> |
|---------|---|

Description

Link function for the exponential family.

Usage

```
linkfcn(mu, linkp, family = "gaussian")
```

```
linkinv(z, linkp, family = "gaussian")
```

Arguments

| | |
|--------|---|
| mu | Numeric. The mean of the response variable. |
| linkp | The link function parameter. A scalar but for the binomial family is also allowed to have the character values "logit" or "probit". |
| family | The distribution of the response variable. |
| z | Numeric. The linear predictor. |

Details

linkfcn maps the mean of the response variable mu to the linear predictor z. linkinv is its inverse. Note that the logit link for the binomial family is defined as the quantile of the logistic distribution with scale 0.6458.

For the Gaussian family, if the link parameter is positive, then the extended link is used, defined by

$$z = \frac{\text{sign}(\mu)|\mu|^\nu - 1}{\nu}$$

In the other case, the usual Box-Cox link is used.

For the Poisson and gamma families, if the link parameter is positive, then the link is defined by

$$z = \frac{\text{sign}(w)(e^{\nu|w|} - 1)}{\nu}$$

where $w = \log(\mu)$. In the other case, the usual Box-Cox link is used.

For the GEV binomial family, the link function is defined by

$$\mu = 1 - \exp\{-\max(0, 1 + \nu z)^{\frac{1}{\nu}}\}$$

for any real ν . At $\nu = 0$ it reduces to the complementary log-log link.

The Wallace binomial family is a fast approximation to the robit family. It is defined as

$$\mu = \Phi(\text{sign}(z)c(\nu)\sqrt{\nu \log(1 + z^2/\nu)})$$

where $c(\nu) = (8\nu + 1)/(8\nu + 3)$

Value

A numeric array of the same dimension as the function's first argument.

See Also

[comparebinlinks](#)

Examples

```
## Not run:
mu <- seq(0.1, 0.9, 0.1)
linkfcn(mu, 7, "binomial")      # robit(7) link function
linkfcn(mu, "logit", "binomial") # logit link function

mu <- seq(-3, 3, 1)
linkfcn(mu, 0.5, "gaussian")    # sqrt transformation
linkinv(linkfcn(mu, 0.5, "gaussian"), 0.5, "gaussian")
curve(linkfcn(x, 0.5, "gaussian"), -3, 3)

## End(Not run)
```

mcmcmake

Convert to an [mcmc](#) object

Description

Convert to an [mcmc](#) object.

Usage

```
mcmcmake(...)
```

Arguments

... Output(s) from the functions mentioned in the Details.

Details

This function takes as input the one or more output(s) from function [mcsglmm](#) or [mcstrga](#) and returns an [mcmc](#) object or an [mcmc.list](#) object for coda. The function requires the coda package to be installed. The spatial random field components are assigned the names `z_*` where `*` is a number beginning at 1. Similarly, the regressor coefficients are assigned the names `beta_*` if not unique, or simply `beta` if there is only one regressor. The names `ssq`, `tsq`, `phi`, `omg` correspond to the partial sill, measurement error variance, spatial range, and relative nugget parameters respectively.

Value

An [mcmc](#) object.

See Also

Functions such as `plot.mcmc` and `summary.mcmc` in the coda package. The function `do.call` can be used to pass arguments stored in a list.

Examples

```
## Not run:
### Load the data
data(rhizoctonia)
rhiz <- na.omit(rhizoctonia)
rhiz$IR <- rhiz$Infected/rhiz$Total # Incidence rate of the
                                     # rhizoctonia disease

### Define the model
corrf <- "spherical"
ssqdf <- 1
ssqsc <- 1
tsqdf <- 1
tsqsc <- 1
betm0 <- 0
betQ0 <- diag(.01, 2, 2)
phiprior <- c(200, 1, 1000, 100) # U(100, 300)
phisc <- 1
omgprior <- c(3, 1, 1000, 0) # U(0, 3)
omgsc <- 1.3
linkp <- 1

## MCMC parameters
Nout <- 100
Nbi <- 0
Nthin <- 1

### Run MCMC
sample <- mcstrga(Yield ~ IR, data = rhiz,
                 atsample = ~ Xcoord + Ycoord, corrf = corrf,
                 Nout = Nout, Nthin = Nthin,
                 Nbi = Nbi, betm0 = betm0, betQ0 = betQ0,
                 ssqdf = ssqdf, ssqsc = ssqsc,
                 tsqdf = tsqdf, tsqsc = tsqsc,
                 phipars = phiprior, omgpars = omgprior,
                 linkp = linkp,
                 phisc = phisc, omgsc = omgsc, test=FALSE)

mcsample <- mcmcmake(sample)
plot(mcsample[, c("phi", "omg", "beta_1", "beta_2", "ssq", "tsq")],
     density = FALSE)
summary(mcsample[, c("phi", "omg", "beta_1", "beta_2", "ssq", "tsq")])

## End(Not run)
```

mcsglmm

*MCMC samples from the Spatial GLMM***Description**

Draw MCMC samples from the Spatial GLMM with known link function

Usage

```
mcsglmm(formula, family = "gaussian", data, weights, subset, atsample, Nout,
         Nthin = 1, Nbi = 0, betm0, betQ0, ssqdf, ssqsc, phipars, omgpars,
         corrfcn = "matern", kappa, linkp, phisc, omgsc, phistart, omgstart,
         dispersion = 1, longlat = FALSE, test = FALSE)
```

Arguments

| | |
|----------|---|
| formula | A representation of the model in the form response ~ terms. The response must be set to NA's at the prediction locations (see the examples on how to do this using the function stackdata). At the observed locations the response is assumed to be a total of replicated measurements. The number of replications is inputted using the argument weights. |
| family | The distribution of the data. The "GEVbinomial" family is the binomial family with link the GEV link (see Details). |
| data | An optional data frame containing the variables in the model. |
| weights | An optional vector of weights. Number of replicated samples for Gaussian and gamma, number of trials for binomial, time length for Poisson. |
| subset | An optional vector specifying a subset of observations to be used in the fitting process. |
| atsample | A formula in the form ~ x1 + x2 + ... + xd with the coordinates of the sampled locations. |
| Nout | Number of MCMC samples to return. This can be a vector for running independent chains. |
| Nthin | The thinning of the MCMC algorithm. |
| Nbi | The burn-in of the MCMC algorithm. |
| betm0 | Prior mean for beta (a vector or scalar). |
| betQ0 | Prior standardised precision (inverse variance) matrix. Can be a scalar, vector or matrix. The first two imply a diagonal with those elements. Set this to 0 to indicate a flat improper prior. |
| ssqdf | Degrees of freedom for the scaled inverse chi-square prior for the partial sill parameter. |
| ssqsc | Scale for the scaled inverse chi-square prior for the partial sill parameter. |
| phipars | Parameters for the generalized inverse gamma prior for the spatial range parameter phi. A four dimensional vector with parameters scale, shape, exponent, location in that order. See Details. |

| | |
|------------|---|
| omgpars | Parameters for the generalized inverse gamma prior for the relative nugget parameter omg . A four dimensional vector with parameters scale, shape, exponent, location in that order. See Details. |
| corrfcn | Spatial correlation function. See geoBayes_correlation for details. |
| kappa | Spatial correlation parameter. Smoothness parameter for Matern, exponent for the power family. |
| linkp | Parameter of the link function. For binomial, a positive number for the degrees of freedom of the robit family or "logit" or "probit". For the other families any number for the exponent of the Box-Cox transformation. |
| phisc | Random walk parameter for ϕ . Smaller values increase the acceptance ratio. Set this to 0 for fixed ϕ . In this case the fixed value is given in the argument <code>phistart</code> . |
| omgsc | Random walk parameter for omg . Smaller values increase the acceptance ratio. Set this to 0 for fixed omg . In this case the fixed value is given in the argument <code>omgstart</code> . |
| phistart | Optional starting value for the MCMC for the spatial range parameter ϕ . Defaults to the mean of its prior. If <code>phisc</code> is 0, then this argument is required and it corresponds to the fixed value of ϕ . This can be a vector of the same length as <code>Nout</code> . |
| omgstart | Optional starting value for the MCMC for the relative nugget parameter omg . Defaults to the mean of its prior. If <code>omgsc</code> is 0, then this argument is required and it corresponds to the fixed value of omg . This can be a vector of the same length as <code>Nout</code> . |
| dispersion | The fixed dispersion parameter. |
| longlat | How to compute the distance between locations. If FALSE, Euclidean distance, if TRUE Great Circle distance. See spDists . |
| test | Whether this is a trial run to monitor the acceptance ratio of the random walk for ϕ and omg . If set to TRUE, the acceptance ratio will be printed on the screen every 100 iterations of the MCMC. Tune the <code>phisc</code> and <code>omgsc</code> parameters in order to achive 20 to 30% acceptance. Set this to a positive number to change the default 100. No thinning or burn-in are done when testing. |

Details

The four-parameter prior for ϕ is defined by

$$\propto (\phi - \theta_4)^{\theta_2 - 1} \exp\left\{-\left(\frac{\phi - \theta_4}{\theta_1}\right)^{\theta_3}\right\}$$

for $\phi > \theta_4$. The prior for omg is similar. The prior parameters correspond to scale, shape, exponent, and location. See [arXiv:1005.3274](#) for details of this distribution.

The GEV (Generalised Extreme Value) link is defined by

$$\mu = 1 - \exp\left\{-\max(0, 1 + \nu x)^{\frac{1}{\nu}}\right\}$$

for any real ν . At $\nu = 0$ it reduces to the complementary log-log link.

Value

A list containing the MCMC samples and other variables as follows:

- `z` A matrix containing the MCMC samples for the spatial random field. Each column is one sample.
- `mu` A matrix containing the MCMC samples for the mean response (a transformation of `z`). Each column is one sample.
- `beta` A matrix containing the MCMC samples for the regressor coefficients. Each column is one sample.
- `ssq` A vector with the MCMC samples for the partial sill parameter.
- `phi` A vector with the MCMC samples for the spatial range parameter.
- `omg` A vector with the MCMC samples for the relative nugget parameter.
- `nu` The link function parameter translated to numeric code used internally.
- `logLik` A vector containing the value of the log-likelihood evaluated at each sample.
- `acc_ratio` The acceptance ratio for the joint update of the parameters `phi` and `omg`.
- `sys_time` The total computing time for the MCMC sampling.
- `Nout`, `Nbi`, `Nthin` As in input. Used internally in other functions.
- `response` The value of the response variable at the observed locations. Used internally in other functions.
- `weights` The response weights at the observed locations. Used internally in other functions.
- `modelmatrix` The model matrix at the observed locations. Used internally in other functions.
- `family` As in input. Used internally in other functions.
- `betm0`, `betQ0`, `ssqdf`, `ssqsc`, `corrfcn`, `kappa`, `dispersion` As in input. Used internally in other functions.
- `locations` Coordinates of the observed locations. Used internally in other functions.
- `whichobs` A logical vector indicated which rows in the data and in the MCMC samples for the spatial random field correspond to the observed locations.

Examples

```
## Not run:
data(rhizoctonia)

### Create prediction grid
predgrid <- mkpredgrid2d(rhizoctonia[c("Xcoord", "Ycoord")],
                        par.x = 100, chull = TRUE, exf = 1.2)

### Combine observed and prediction locations
rhizdata <- stackdata(rhizoctonia, predgrid$grid)

### Define the model
corrfunc <- "spherical"
kappa <- 0
ssqdf <- 1
```

```

ssqsc <- 1
betm0 <- 0
betQ0 <- .01
hiprior <- c(100, 1, 1000, 100) # U(100, 200)
phisc <- 3
omgprior <- c(2, 1, 1, 0)      # Exp(mean = 2)
omgsc <- .1
linkp <- "probit"

### MCMC sizes
Nout <- 100
Nthin <- 1
Nbi <- 0

### Trial run
emt <- mcsglmm(Infected ~ 1, 'binomial', rhizdata, weights = Total,
              atsample = ~ Xcoord + Ycoord,
              Nout = Nout, Nthin = Nthin, Nbi = Nbi,
              betm0 = betm0, betQ0 = betQ0, ssqdf = ssqdf, ssqsc = ssqsc,
              phipars = hiprior, omgpars = omgprior, linkp = linkp,
              corrfcn = corrf, kappa = kappa, phisc = phisc, omgsc = omgsc,
              dispersion = 1, test = 10)

### Full run
emc <- mcsglmm(Infected ~ 1, 'binomial', rhizdata, weights = Total,
              atsample = ~ Xcoord + Ycoord,
              Nout = Nout, Nthin = Nthin, Nbi = Nbi,
              betm0 = betm0, betQ0 = betQ0, ssqdf = ssqdf, ssqsc = ssqsc,
              phipars = hiprior, omgpars = omgprior, linkp = linkp,
              corrfcn = corrf, kappa = kappa, phisc = phisc, omgsc = omgsc,
              dispersion = 1, test = FALSE)

plot.ts(cbind(phi = emc$phi, omg = emc$omg, beta = c(emc$beta),
             ssq = emc$ssq), nc = 2)

emcmc <- mcmcmake(emc)
summary(emcmc[, c("phi", "omg", "beta", "ssq")])

## End(Not run)

```

mcstrga

MCMC samples from the transformed Gaussian model

Description

Draw MCMC samples from the transformed Gaussian model with known link function

Usage

```
mcstrga(formula, data, weights, subset, atsample, Nout, Nthin = 1, Nbi = 0,
        betm0, betQ0, ssqdf, ssqsc, tsqdf, tsqsc, phipars, omgpars,
        corrfcn = "matern", kappa, linkp, phisc, omgsc, phistart, omgstart,
        longlat = FALSE, test = FALSE)
```

Arguments

| | |
|----------|---|
| formula | A representation of the model in the form response ~ terms. The response must be set to NA's at the prediction locations (see the example in mcsglmm for how to do this using stackdata). At the observed locations the response is assumed to be a total of replicated measurements. The number of replications is inputted using the argument weights. |
| data | An optional data frame containing the variables in the model. |
| weights | An optional vector of weights. Number of replicated samples. |
| subset | An optional vector specifying a subset of observations to be used in the fitting process. |
| atsample | A formula in the form ~ x1 + x2 + ... + xd with the coordinates of the sampled locations. |
| Nout | Number of MCMC samples to return. This can be a vector for running independent chains. |
| Nthin | The thinning of the MCMC algorithm. |
| Nbi | The burn-in of the MCMC algorithm. |
| betm0 | Prior mean for beta (a vector or scalar). |
| betQ0 | Prior standardised precision (inverse variance) matrix. Can be a scalar, vector or matrix. The first two imply a diagonal with those elements. Set this to 0 to indicate a flat improper prior. |
| ssqdf | Degrees of freedom for the scaled inverse chi-square prior for the partial sill parameter. |
| ssqsc | Scale for the scaled inverse chi-square prior for the partial sill parameter. |
| tsqdf | Degrees of freedom for the scaled inverse chi-square prior for the measurement error parameter. |
| tsqsc | Scale for the scaled inverse chi-square prior for the measurement error parameter. |
| phipars | Parameters for the generalized inverse gamma prior for the range parameter phi. A four dimensional vector with parameters scale, shape, exponent, location in that order. See mcsglmm . |
| omgpars | Parameters for the generalized inverse gamma prior for the relative nugget parameter omg. A four dimensional vector with parameters scale, shape, exponent, location in that order. See mcsglmm . |
| corrfcn | Spatial correlation function. See geoBayes_correlation for details. |
| kappa | Spatial correlation parameter. Smoothness parameter for Matern, exponent for the power family. |

| | |
|----------|--|
| linkp | The exponent of the Box-Cox transformation. |
| phisc | Random walk parameter for ϕ . Smaller values increase the acceptance ratio. Set this to 0 for fixed ϕ . In this case the fixed value is given in the argument <code>phistart</code> . |
| omgsc | Random walk parameter for ω . Smaller values increase the acceptance ratio. Set this to 0 for fixed ω . In this case the fixed value is given in the argument <code>omgstart</code> . |
| phistart | Optional starting value for the MCMC for the spatial range parameter ϕ . Defaults to the mean of its prior. If <code>phisc</code> is 0, then this argument is required and it corresponds to the fixed value of ϕ . This can be a vector of the same length as <code>Nout</code> . |
| omgstart | Optional starting value for the MCMC for the relative nugget parameter ω . Defaults to the mean of its prior. If <code>omgsc</code> is 0, then this argument is required and it corresponds to the fixed value of ω . This can be a vector of the same length as <code>Nout</code> . |
| longlat | How to compute the distance between locations. If FALSE, Euclidean distance, if TRUE Great Circle distance. See spDists . |
| test | Whether this is a trial run to monitor the acceptance ratio of the random walk for ϕ and ω . If set to TRUE, the acceptance ratio will be printed on the screen every 100 iterations of the MCMC. Tune the <code>phisc</code> and <code>omgsc</code> parameters in order to achieve 20 to 30% acceptance. Set this to a positive number to change the default 100. No thinning or burn-in are done when testing. |

Details

Simulates from the posterior distribution of this model.

Value

A list containing the MCMC samples and other variables as follows:

- `z` A matrix containing the MCMC samples for the spatial random field. Each column is one sample.
- `mu` A matrix containing the MCMC samples for the mean response (a transformation of `z`). Each column is one sample.
- `beta` A matrix containing the MCMC samples for the regressor coefficients. Each column is one sample.
- `ssq` A vector with the MCMC samples for the partial
- `tsq` A vector with the MCMC samples for the measurement error variance.
- `phi` A vector with the MCMC samples for the spatial range parameter.
- `omg` A vector with the MCMC samples for the relative nugget parameter.
- `nu` The link function parameter translated to numeric code used internally.
- `logLik` A vector containing the value of the log-likelihood evaluated at each sample.
- `acc_ratio` The acceptance ratio for the joint update of the parameters ϕ and ω .

- `sys_time` The total computing time for the MCMC sampling.
- `Nout`, `Nbi`, `Nthin` As in input. Used internally in other functions.
- `response` The average of the response variable at the observed locations, i.e. its value divided by the corresponding weight. Used internally in other functions.
- `weights` The response weights at the observed locations. Used internally in other functions.
- `modelmatrix` The model matrix at the observed locations. Used internally in other functions.
- `family` As in input. Used internally in other functions.
- `betm0`, `betQ0`, `ssqdf`, `ssqsc`, `corrfcn`, `kappa`, `tsqdf`, `tsqsc` As in input. Used internally in other functions.
- `locations` Coordinates of the observed locations. Used internally in other functions.
- `whichobs` A logical vector indicated which rows in the data and in the MCMC samples for the spatial random field correspond to the observed locations.

Examples

```
## Not run:
### Load the data
data(rhizoctonia)
rhiz <- na.omit(rhizoctonia)
rhiz$IR <- rhiz$Infected/rhiz$Total # Incidence rate of the
                                     # rhizoctonia disease

### Define the model
corrff <- "spherical"
ssqdf <- 1
ssqsc <- 1
tsqdf <- 1
tsqsc <- 1
betm0 <- 0
betQ0 <- diag(.01, 2, 2)
phiprior <- c(200, 1, 1000, 100) # U(100, 300)
phisc <- 1
omgprior <- c(3, 1, 1000, 0) # U(0, 3)
omgsc <- 1.3
linkp <- 1

## MCMC parameters
Nout <- 100
Nbi <- 0
Nthin <- 1

samplt <- mcstrga(Yield ~ IR, data = rhiz,
                 atsample = ~ Xcoord + Ycoord, corrff = corrff,
                 Nout = Nout, Nthin = Nthin,
                 Nbi = Nbi, betm0 = betm0, betQ0 = betQ0,
                 ssqdf = ssqdf, ssqsc = ssqsc,
                 tsqdf = tsqdf, tsqsc = tsqsc,
                 phipars = phiprior, omgpars = omgprior,
                 linkp = linkp,
```



```

      phisc = phisc, omgsc = omgsc, test=10)

sample <- mcstrga(Yield ~ IR, data = rhiz,
  atsample = ~ Xcoord + Ycoord, corrf = corrf,
  Nout = Nout, Nthin = Nthin,
  Nbi = Nbi, betm0 = betm0, betQ0 = betQ0,
  ssqdf = ssqdf, ssqsc = ssqsc,
  tsqdf = tsqdf, tsqsc = tsqsc,
  phipars = phiprior, omgpars = omgprior,
  linkp = linkp,
  phisc = phisc, omgsc = omgsc, test=FALSE)

## End(Not run)

```

mkpredgrid2d

Make prediction grid

Description

This function creates a grid for prediction.

Usage

```

mkpredgrid2d(pnts.x, pnts.y, par.x, par.y, isby = FALSE, chull = FALSE,
  exf = 1)

```

Arguments

| | |
|--------|--|
| pnts.x | x coordinate of the domain. Could also be a two-column matrix containing the x and y coordinates |
| pnts.y | y coordinate of the domain. Should be omitted or set to NULL if the argument pnts.x is a two-column matrix. |
| par.x | A scalar parameter for the x component of the new grid. This parameter corresponds to either the by or the length.out arguments of the function seq . Could also be a vector of two elements containing the parameter for x and y. |
| par.y | As in par.x for the y component of the new grid. Should be omitted or set to NULL if the argument par.x is a two-dimensional vector. |
| isby | If TRUE, the arguments par.x and par.y correspond to the by argument of the function seq , otherwise they correspond to length.out. |
| chull | Whether to calculate the convex hull of the points. Set this to TRUE if pnts.x and pnts.y denote the sampled locations. If they correspond to the borders of the domain, it is recommended to set this to FALSE. |
| exf | An expansion factor of the convex hull of cbind(pnts.x, pnts.y). Must be positive. If larger or smaller than 1, the convex hull is respectively expanded or contracted. |

Details

If `chull` this function first calculates the convex hull of the points. If `exf` is not 1 the borders are expanded. Then the function calls `point.in.polygon` to select points that fall inside the borders.

Value

A list with components

- `grid` A two-column matrix with the prediction grid
- `xycoord` A list with components "x" and "y" containing the sequence of points used to create the grid
- `xygrid` A matrix with the full square grid derived from `xycoord`
- `borders` The (expanded) borders of the domain

See Also

`pred_grid`

Examples

```
## Not run:
data(rhizoctonia)
predgrid <- mkpredgrid2d(rhizoctonia[c("Xcoord", "Ycoord")],
                        par.x = 100, chull = TRUE, exf = 1.2)
plot(predgrid$borders, type = "l") # Domain for prediction
points(predgrid$grid, pch = 20, cex = .3) # Prediction locations
points(rhizoctonia[c("Xcoord", "Ycoord")]) # Observed locations

## End(Not run)
```

plotbf2

Plot the estimated Bayes factors

Description

This function plots the estimated logarithm Bayes factors from the function `bf2new`.

Usage

```
plotbf2(bf2obj, pars = c("linkp", "phi", "omg", "kappa"),
        profile = length(pars) > 2, ...)
```

Arguments

| | |
|----------------------|---|
| <code>bf2obj</code> | Output from the function <code>bf2new</code> . |
| <code>pars</code> | A vector with the names of the parameters to plot. |
| <code>profile</code> | Whether it should produce a profile plot or a contour plot if the length of <code>pars</code> is 2. |
| <code>...</code> | Other input to be passed to either <code>plot</code> or <code>contour</code> . |

Details

Depending on whether pars has length 1 or 2, this function creates a line or a contour plot of the estimated Bayes factors. If its length is 3 or 4, then it produces multiple profile plots. In this case the variable is fixed at different values and the maximum Bayes factor corresponding to the fixed value is plotted against that value.

Value

This function returns nothing.

Examples

```
## Not run:
data(rhizoctonia)
### Define the model
corrff <- "spherical"
kappa <- 0
ssqdf <- 1
ssqsc <- 1
betm0 <- 0
betQ0 <- .01
linkp <- "probit"
### Skeleton points
philist <- c(100, 140, 180)
omglist <- c(.5, 1)
parlist <- expand.grid(phi=philist, linkp=linkp, omg=omglist, kappa = kappa)
### MCMC sizes
Nout <- 100
Nthin <- 1
Nbi <- 0
### Take MCMC samples
runs <- list()
for (i in 1:NROW(parlist)) {
  runs[[i]] <- mcsglmm(Infected ~ 1, 'binomial', rhizoctonia, weights = Total,
    atsample = ~ Xcoord + Ycoord,
    Nout = Nout, Nthin = Nthin, Nbi = Nbi,
    betm0 = betm0, betQ0 = betQ0,
    ssqdf = ssqdf, ssqsc = ssqsc,
    phistart = parlist$phi[i], omgstart = parlist$omg[i],
    linkp = parlist$linkp[i], kappa = parlist$kappa[i],
    corrfcn = corrff, phisc = 0, omgsc = 0)
}
bf <- bf1skel(runs)
bfall <- bf2new(bf, phi = seq(100, 200, 10), omg = seq(0, 2, .2))
plotbf2(bfall, c("phi", "omg"))
plotbf2(bfall, c("phi", "omg"), profile = TRUE, type = "b", ylab="log(BF)")

## End(Not run)
```

| | |
|-----------|---|
| revlogreg | <i>Reverse logistic regression estimation</i> |
|-----------|---|

Description

Perform the reverse logistic regression estimation

Usage

```
revlogreg(lgk, N)
```

Arguments

| | |
|-----|---|
| lgk | The value of the loglikelihood at different samples and different parameters. This should be entered as a matrix where the rows are the values of the samples and the columns correspond to the parameters. The [i,j] element of the matrix is the value of the loglikelihood at the ith sample when all samples are put together evaluated at the jth parameter value. |
| N | A vector of length ncol(lgk) or a scalar corresponding to the sample sizes from each model. Must $\text{sum}(N) == \text{nrow}(lgk)$. The first N[1] samples come from model corresponding to the first set of parameters, then (N[1]+1):N[2] are from the model corresponding to the second set of parameters, and so on. |

Details

Estimation is done by maximising the reverse logistic log likelihood.

Value

A vector containing the reverse logistic regression estimates of the logarithm of the Bayes factors. The first set of parameters is taken as the reference model so its estimate is always 0.

| | |
|-------------|--|
| rhizoctonia | <i>Rhizoctonia root rot infections</i> |
|-------------|--|

Description

Rhizoctonia root rot infections.

Usage

```
data(rhizoctonia)
```

Format

A data frame with 100 rows and 5 variables.

Details

A dataset containing the number of infected roots and the sample coordinate. The data were collected by Dr Jim Cook at Washington State University.

- Xcoord Longitude of the sampling location.
- Ycoord Latitude of the sampling location.
- Total Total number of roots sampled at that location.
- Infected Number of infected roots found at that location.
- Yield Barley yield at that location. These data were obtained by hand-harvesting a 4-square-meter area in the sampling location. One observation is missing.

Note

We acknowledge Hao Zhang for providing these data.

Source

www.biometrics.tibs.org/datasets/010434.txt

References

Zhang, H. (2002). On estimation and prediction for spatial generalized linear mixed models. *Biometrics*, 58(1), 129-136.

rsglmm

Simulation from a spatial model

Description

Simulate from a variety of spatial models.

Usage

```
rsglmm(n, formula, family = "gaussian", data, weights, subset, atsample, beta,
linkp, phi, omg, kappa, ssq, corrfcn = "matern", longlat = FALSE,
dispersion = 1, returnGRF = FALSE, warndisp = TRUE)
```

```
rstrga(n, formula, data, weights, subset, atsample, beta, linkp, phi, omg,
kappa, ssq, corrfcn = "matern", longlat = FALSE, dispersion = 1,
returnGRF = FALSE)
```

```
rsgrf(n, formula, data, subset, atsample, beta, phi, omg, kappa, ssq,
corrfcn = "matern", longlat = FALSE)
```

Arguments

| | |
|------------|---|
| n | The number of instances to simulate |
| formula | A representation of the model in the form response ~ terms. The LHS can be omitted. If the LHS exists, it can be of the form y, y z, or sums of terms at either side of the to specify the names of the variables to include in the data frame. |
| family | The distribution of the data to simulate from. |
| data | An optional data frame containing the variables in the model. |
| weights | An optional vector of weights. Number of replicated samples for Gaussian and gamma, number of trials for binomial, time length for Poisson. |
| subset | An optional set of indices. Simulations will be provided for those locations only. |
| atsample | A formula of the form ~ Xcoord + Ycoord specifying the sampled locations. |
| beta | A vector of the regressor coefficients to use. |
| linkp | The link function parameter. |
| phi | The spatial range parameter. |
| omg | The relative nugget parameter. |
| kappa | The spatial smoothness parameter. |
| ssq | The partial sill parameter. |
| corrfcn | The correlation function to use. |
| longlat | How to compute the distance between locations. If FALSE, Euclidean distance, if TRUE Great Circle distance. See spDists . |
| dispersion | The fixed dispersion parameter. When this is not 1 and the sample is from a binomial or a Poisson distribution, an approximate sample is returned. |
| returnGRF | Whether to return the simulate Gaussian random field as well. |
| warndisp | Whether to warn when sampling from a quasi distribution. This is the case for binomial and Poisson when the dispersion is not one. |

Details

The spatial Gaussian random field is simulated using the Cholesky decomposition of the covariance matrix.

The sample from a quasi distribution uses a hack which matches the mean and the variance of the distribution. See the source code for details.

Value

A data frame containing the predictors, sampling locations, optional weights, and samples.

Examples

```
## Not run:
n <- 100
beta <- c(-2, 1)
phi <- .2
omg <- .3
linkp <- 0
ssq <- 1
l <- rep(10, n)
corrf <- "matern"
kappa <- .5
family <- "poisson"
Xcoord <- runif(n)
Ycoord <- runif(n)
f <- Xcoord + Ycoord
formula <- y|z ~ f
mydata <- rsglmm(1, formula, family, weights = 1,
  atsample = ~ Xcoord + Ycoord, beta = beta, linkp = linkp,
  phi = phi, omg = omg, kappa = kappa, ssq = ssq,
  corrfcn = corrf, returnGRF = TRUE)

## End(Not run)
```

sploglik

Spatial log likelihood

Description

Spatial joint log likelihood

Usage

```
sploglik(pargrid, runs, transf = c("no", "mu", "wo"))
```

Arguments

| | |
|---------|--|
| pargrid | A data frame with components "linkp", "phi", "omg", "kappa". Each row gives a combination of the parameters to compute the log-likelihood. |
| runs | A list with outputs from the function <code>mcsglmm</code> or <code>mcstrga</code> . |
| transf | Whether to use a transformed sample for the computations. If "no" or FALSE, it doesn't. If "mu" or TRUE, it uses the samples for the mean. If "wo" it uses an alternative transformation. The latter can be used only for the families indicated by <code>.geoBayes_models\$haswo</code> . |

Details

Computes the joint log likelihood $\log f(y,z|\text{parameters})$ or $\log f(y,\mu|\text{parameters})$ for each (y,z) or (y,μ) in runs and for parameters in pargrid up to a constant which does not depend on the parameters.

Value

A matrix with number of rows the total number of samples in runs and number of columns the number of rows in pargrid. The [i,j] element of the matrix is the value of the loglikelihood at the ith sample when all samples in runs are put together evaluated at the jth parameter value.

| | |
|-----------|----------------------------|
| stackdata | <i>Combine data.frames</i> |
|-----------|----------------------------|

Description

Combine data.frames

Usage

```
stackdata(..., fillwith = NA, keepclass = FALSE)
```

Arguments

| | |
|-----------|---|
| ... | data.frames or objects that can be coerced to data.frames |
| fillwith | Which value to use for missing variables. This could be a scalar, a named vector, or a named list with one value in each component; see Details. |
| keepclass | Whether to preserve the <code>class</code> of each variable. The elements in <code>fillwith</code> are coerced to the corresponding variable's class. |

Details

This function combines data.frames by filling in missing variables. This is useful for combining data from sampled locations with prediction locations.

If `fillwith` is a named object, its names must correspond to the names of variables in the data frames. If a variable is missing, then it is filled with the corresponding value in `fillwith`. `fillwith` can contain only one unnamed component which corresponds to the default filling.

Value

A stacked data.frame.

Examples

```
## Not run:
d1 <- data.frame(w = 1:3, z = 4:6 + 0.1)
d2 <- data.frame(w = 3:7, x = 1:5, y = 6:10)
(d12a <- stackdata(d1, d2))
lapply(d12a, class)
(d12b <- stackdata(d1, d2, fillwith = c(x = NA, y = 0, z = -99)))
lapply(d12b, class)
(d12c <- stackdata(d1, d2, fillwith = c(x = NA, y = 0, z = -99),
  keepclass = TRUE))
```



```

lapply(d12c, class)
(d12d <- stackdata(d1, d2, fillwith = c(x = NA, 0)))

data(rhizoctonia)
predgrid <- mkpredgrid2d(rhizoctonia[c("Xcoord", "Ycoord")],
                        par.x = 100, chull = TRUE, exf = 1.2)
rhizdata <- stackdata(rhizoctonia, predgrid$grid)

## End(Not run)

```

| | |
|----------------|--------------------------|
| subset.geomcmc | <i>Subset MCMC chain</i> |
|----------------|--------------------------|

Description

Return subset of MCMC chain.

Usage

```

## S3 method for class 'geomcmc'
subset(x, subset, ...)

```

Arguments

| | |
|--------|--|
| x | Output from the functions <code>mcsglmm</code> or <code>mcstrga</code> . |
| subset | Logical or integer vector. |
| ... | Further arguments to be passed to or from other methods. |

Value

A similar object as x with the subsetted chain.

| | |
|--------------|-----------------------------------|
| transfsample | <i>Compute transformed sample</i> |
|--------------|-----------------------------------|

Description

Compute transformed sample.

Usage

```

transfsample(runs, model, transf = c("no", "mu", "wo"))

```

Arguments

| | |
|--------|--|
| runs | A list of lists with elements z , μ , ν , whichobs which gives the samples and link parameter. |
| model | A list with elements response and family. |
| transf | The type of transformation to use. |

Details

Computes the transformed sample according to transf.

Value

A list with elements sample, transf, itr, ifam.

Index

*Topic **datasets**

geoBayes_correlation, [11](#)
geoBayes_models, [12](#)

*Topic **package**

geoBayes, [10](#)
.geoBayes_corrfcn, [13](#)
.geoBayes_corrfcn
(geoBayes_correlation), [11](#)
.geoBayes_models, [12](#)
.geoBayes_models (geoBayes_models), [12](#)

bf1skel, [2](#), [4](#), [6](#)
bf2new, [3](#), [4](#), [6](#), [26](#)
bf2optim, [3](#), [6](#)
bf2se, [7](#)
bfse, [8](#)

class, [32](#)
comparebinlinks, [9](#), [16](#)

do.call, [17](#)

geoBayes, [10](#)
geoBayes-package (geoBayes), [10](#)
geoBayes_correlation, [11](#), [14](#), [19](#), [22](#)
geoBayes_models, [12](#)

likaprxn, [12](#)
likoptim, [13](#)
linkfcn, [15](#)
linkinv (linkfcn), [15](#)

mcmc, [16](#)
mcmc.list, [16](#)
mcmcmake, [16](#)
mcsglmm, [2](#), [8](#), [16](#), [18](#), [22](#), [31](#), [33](#)
mcstrga, [2](#), [8](#), [16](#), [21](#), [31](#), [33](#)
mkpredgrid2d, [25](#)

optim, [6](#), [14](#)

plot.mcmc, [17](#)
plotbf2, [26](#)
point.in.polygon, [26](#)
pred_grid, [26](#)

revlogreg, [28](#)
rhizoctonia, [28](#)
rsglmm, [29](#)
rsgrf (rsglmm), [29](#)
rstrga (rsglmm), [29](#)

seq, [25](#)
spDists, [13](#), [14](#), [19](#), [23](#), [30](#)
sploglik, [31](#)
stackdata, [18](#), [22](#), [32](#)
subset.geomcmc, [33](#)
summary.mcmc, [17](#)

transfsample, [33](#)