Package 'geoBayes'

January 26, 2026

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 0.7.5

Date 2026-01-26

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

Imports coda, graphics, sp, stats, optimx

Depends R (>= 3.0.0)

LazyData true

Encoding UTF-8

License GPL (>= 2)

Classification/ACM 86A32, 62M30, 62F15

ByteCompile yes

NeedsCompilation yes

RoxygenNote 7.3.1

Author Evangelos Evangelou [aut, cre], Vivekananda Roy [aut]

Repository CRAN

Date/Publication 2026-01-26 11:40:15 UTC

Contents

`alik_cutoff` *Approximate log-likelihood calculation*

Description

Calculate the likelihood approximation at different parameter values. This function is useful for choosing the skeleton set.

Plot likelihood approximation.

Usage

```
alik_cutoff(likopt, par_vals, likthreshold)

alik_plot(alikobj)
```

Arguments

likopt	Output from the function alik_optim.
par_vals	A named list with some of the components "linkp", "phi", "omg", "kappa".
likthreshold	A threshold value proportion to calculate the cutoff. The cutoff will be calculated as that proportion relative to the maximum value of the log-likelihood.
alikobj	Output from alik_cutoff.

Details

The input par_vals is meant to contain vector of parameter values for each parameter. For each element in par_vals, the other parameters are set equal to the maximisers given in likopt and the approximate likelihood is computed. The cuttoff is calculated using linear interpolation provided by approx.

The plot can be used to visualise the Laplace approximation to the likelihood provided by the function alik_cutoff.

Value

A list with the log-likelihood approximation and cutoff values.

Draws a plot.

References

Evangelou, E., & Roy, V. (2019). Estimation and prediction for spatial generalized linear mixed models with parametric links via reparameterized importance sampling. Spatial Statistics, 29, 289-315.

---

alik_inla *Log-likelihood approximation*

Description

Log-likelihood approximation.

Usage

```
alik_inla(
  par_vals,
  formula,
  family = "gaussian",
  data,
  weights,
  subset,
  offset,
  atsample,
```

```
  corrfcn = "matern",
  np,
  betm0,
  betQ0,
  ssqdf,
  ssqsc,
  tsqdf,
  tsqsc,
  dispersion = 1,
  longlat = FALSE
)
```

Arguments

par_vals	A data frame with the components "linkp", "phi", "omg", "kappa". The approximation will be computed at each row of the data frame.
formula	A representation of the model in the form `response ~ terms`.
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.
offset	See `lm`.
atsample	A formula in the form `~ x1 + x2 + ... + xd` with the coordinates of the sampled locations.
corrfcn	Spatial correlation function. Can be one of the choices in `.geoBayes_corrfcn`.
np	The number of integration points for the spatial variance parameter sigma^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 `spDists`.

Details

Computes and approximation to the log-likelihood for the given parameters using integrated nested Laplace approximations.

Value

A list with components

- `par_vals` A data frame of the parameter values.
- `aloglik` The approximate log-likelihood at thos parameter values.

References

Evangelou, E., & Roy, V. (2019). Estimation and prediction for spatial generalized linear mixed models with parametric links via reparameterized importance sampling. Spatial Statistics, 29, 289-315.

`alik_optim` *Log-likelihood maximisation*

Description

Approximate log-likelihood maximisation

Usage

```
alik_optim(
  paroptim,
  formula,
  family = "gaussian",
  data,
  weights,
  subset,
  offset,
  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. If the compontent'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 `response ~ terms`.
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.
offset	See `lm`.
atsample	A formula in the form `~ x1 + x2 + ... + xd` 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 sigma^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.

References

Evangelou, E., & Roy, V. (2019). Estimation and prediction for spatial generalized linear mixed models with parametric links via reparameterized importance sampling. Spatial Statistics, 29, 289-315.

`bf1skel` *Computation of Bayes factors at the skeleton points*

Description

Function to compute the Bayes factors from MCMC samples.

Usage

```
bf1skel(
  runs,
  bfsize1 = 0.8,
  method = c("RL", "MW"),
  reference = 1,
  transf = c("no", "mu", "wo")
)
```

Arguments

`runs`	A list with outputs from the function `mcsglmm` or `mcstrga`.
`bfsize1`	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 mu 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
corrf <- "spherical"
kappa <- 0
ssqdf <- 1
ssqsc <- 1
betm0 <- 0
betQ0 <- .01
family <- "binomial.probit"
### Skeleton points
philist <- c(100, 140, 180)
omglist <- c(.5, 1)
parlist <- expand.grid(linkp=0, phi=philist, 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, family, rhizoctonia, weights = Total,
                       atsample = ~ Xcoord + Ycoord,
                       Nout = Nout, Nthin = Nthin, Nbi = Nbi,
                       betm0 = betm0, betQ0 = betQ0,
                       ssqdf = ssqdf, ssqsc = ssqsc,
                       phi = parlist$phi[i], omg = parlist$omg[i],
                       linkp = parlist$linkp[i], kappa = parlist$kappa[i],
                       corrfcn = corrf,
                       corrtuning=list(phi = 0, omg = 0, kappa = 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 `length(linkp) * length(phi) * length(omg) * length(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
family <- "binomial.probit"
### Skeleton points
philist <- c(100, 140, 180)
omglist <- c(.5, 1)
parlist <- expand.grid(linkp=0, phi=philist, 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, family, rhizoctonia, weights = Total,
                       atsample = ~ Xcoord + Ycoord,
                       Nout = Nout, Nthin = Nthin, Nbi = Nbi,
                       betm0 = betm0, betQ0 = betQ0,
                       ssqdf = ssqdf, ssqsc = ssqsc,
                       phi = parlist$phi[i], omg = parlist$omg[i],
                       linkp = parlist$linkp[i], kappa = parlist$kappa[i],
                       corrfcn = corrf,
                       corrtuning=list(phi = 0, omg = 0, kappa = 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 compontent'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
corrf <- "spherical"
kappa <- 0
ssqdf <- 1
ssqsc <- 1
betm0 <- 0
betQ0 <- .01
```

```
family <- "binomial.probit"
### Skeleton points
philist <- c(100, 140, 180)
omglist <- c(.5, 1)
parlist <- expand.grid(linkp=0, phi=philist, 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, family, rhizoctonia, weights = Total,
                       atsample = ~ Xcoord + Ycoord,
                       Nout = Nout*c(.8,.2), Nthin = Nthin, Nbi = Nbi,
                       betm0 = betm0, betQ0 = betQ0,
                       ssqdf = ssqdf, ssqsc = ssqsc,
                       phi = parlist$phi[i], omg = parlist$omg[i],
                       linkp = parlist$linkp[i], kappa = parlist$kappa[i],
                       corrfcn = corrf,
                       corrtuning=list(phi = 0, omg = 0, kappa = 0))
}
bf <- bf1skel(runs)
est <- bf2optim(bf, list(phi = c(100, 200), omg = c(0, 2)))
est

## End(Not run)
```

`bf2se` *Empirical Bayes standard errors*

Description

Standard errors for the empirical Bayes estimates of the parameters.

Usage

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

Arguments

`mcrun`	The output from the function `mcsglmm` where the parameters linkp, phi, omg, kappa are set at their empirical Bayes estimates (output of `bf2optim`).
`transf`	The type of transformation to use.

Value

The precision (inverse covariance) matrix.

References

Casella, G. (2001). Empirical Bayes Gibbs sampling. Biostatistics, 2(4), 485-500.

Evangelou, E., & Roy, V. (2019). Estimation and prediction for spatial generalized linear mixed models with parametric links via reparameterized importance sampling. Spatial Statistics, 29, 289-315.

`bmbfse` *Batch means, Bayes factors standard errors*

Description

Compute the standard errors for the Bayes factors estimates.

Usage

```
bmbfse(
  pargrid,
  runs,
  bfsize1 = 0.8,
  nbatch1 = 0.5,
  nbatch2 = 0.5,
  S1method = 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 `mcsglmm` or `mcstrga`.
`bfsize1`	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`.

S1method	Which method to use to calculate the Bayes factors in stage 1: 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 .geoBayes_models$haswo.

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 log Bayes factors and 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.

Roy, V., & Evangelou, E. (2018). Selection of proposal distributions for generalized importance sampling estimators. arXiv preprint arXiv:1805.00829.

---

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 <vroy@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.

Evangelou, E., & Roy, V. (2019). Estimation and prediction for spatial generalized linear mixed models with parametric links via reparameterized importance sampling. *Spatial Statistics*, 29, 289-315.

Roy, V., & Evangelou, E. (2024). Selection of proposal distributions for multiple importance sampling. *Statistica Sinica*, 34, 27-46.

Examples

```
## Not run:
demo(package = "geoBayes")
demo(rhizoctonia1, package = "geoBayes")
demo(rhizoctonia1, 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 5 rows and 4 columns.

geoBayes_models *Models used in the geoBayes package*

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 12 rows and 7 columns.

linkfcn *Calculate the link function for exponential families*

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.
`family`	The distribution of the response variable from `.geoBayes_models`. Either an integer or the family name.
`z`	Numeric. The linear predictor.

Details

`linkfcn` maps the mean of the response variable `mu` to the linear predictor `z`. `linkinv` is its inverse.

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

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

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

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

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.

References

Evangelou, E., & Roy, V. (2019). Estimation and prediction for spatial generalized linear mixed models with parametric links via reparameterized importance sampling. Spatial Statistics, 29, 289-315.

Examples

```
## Not run:
mu <- seq(0.1, 0.9, 0.1)
linkfcn(mu, 7, "binomial")       # robit(7) link function
linkfcn(mu, , "binomial.logit")  # 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,
                  linkp = linkp,
                  corrprior = list(phi = phiprior, omg = omgprior),
                  corrtuning = list(phi = phisc, omg = omgsc, kappa = 0),
                  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,
  offset,
  atsample,
  corrfcn = "matern",
  linkp,
  phi,
```

```
  omg,
  kappa,
  Nout,
  Nthin = 1,
  Nbi = 0,
  betm0,
  betQ0,
  ssqdf,
  ssqsc,
  corrpriors,
  corrtuning,
  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.
offset	See lm.
atsample	A formula in the form `~ x1 + x2 + ... + xd` with the coordinates of the sampled locations.
corrfcn	Spatial correlation function. See geoBayes_correlation for details.
linkp	Parameter of the link function. A scalar value.
phi	Optional starting value for the MCMC for the spatial range parameter `phi`. Defaults to the mean of its prior. If `corrtuning[["phi"]]` is 0, then this argument is required and it corresponds to the fixed value of `phi`. This can be a vector of the same length as Nout.
omg	Optional starting value for the MCMC for the relative nugget parameter `omg`. Defaults to the mean of its prior. If `corrtuning[["omg"]]` 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 Nout.
kappa	Optional starting value for the MCMC for the spatial correlation parameter `kappa` (Matern smoothness or exponential power). Defaults to the mean of its

prior. If `corrtuning[["kappa"]]` is 0 and it is needed for the chosen correlation function, then this argument is required and it corresponds to the fixed value of `kappa`. This can be a vector of the same length as Nout.

`Nout` Number of MCMC samples to return. This can be a vector for running independent chains. 0 elements are dropped.

`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.

`corrpriors` A list with the components `phi`, `omg` and `kappa` as needed. These correspond to the prior distribution parameters. For `phi` and `omg` it must be a vector of length 4. The generalized inverse gamma prior is assumed and the input corresponds to the parameters scale, shape, exponent, location in that order (see Details). For `kappa` it must be a vector of length 2. A uniform prior is assumed and the input corresponds to the lower and upper bounds in that order.

`corrtuning` A vector or list with the components `phi`, `omg` and `kappa` as needed. These correspond to the random walk parameter for the Metropolis-Hastings step. Smaller values increase the acceptance ratio. Set this to 0 for fixed parameter value.

`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 `phi` and `omg`. If set to `TRUE`, the acceptance ratio will be printed on the screen every 100 iterations of the MCMC. Tune the `phisc` and `omgsc` 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 `phi` is defined by

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

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

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

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

Value

A list containing the objects MODEL, DATA, FIXED, MCMC and call. The MCMC samples are stored in the object MCMC 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
- phi A vector with the MCMC samples for the spatial range parameter, if sampled.
- omg A vector with the MCMC samples for the relative nugget parameter, if sampled.
- 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, if sampled.
- sys_time The total computing time for the MCMC sampling.
- Nout, Nbi, Nthin As in input. Used internally in other functions.

The other objects contain input variables. The object call contains the function call.

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
corrf <- "spherical"
family <- "binomial.probit"
kappa <- 0
ssqdf <- 1
ssqsc <- 1
betm0 <- 0
betQ0 <- .01
phiprior <- c(100, 1, 1000, 100) # U(100, 200)
phisc <- 3
omgprior <- c(2, 1, 1, 0)        # Exp(mean = 2)
omgsc <- .1
##'
### MCMC sizes
Nout <- 100
```

```
Nthin <- 1
Nbi <- 0

### Trial run
emt <- mcsglmm(Infected ~ 1, family, rhizdata, weights = Total,
               atsample = ~ Xcoord + Ycoord,
               Nout = Nout, Nthin = Nthin, Nbi = Nbi,
               betm0 = betm0, betQ0 = betQ0, ssqdf = ssqdf, ssqsc = ssqsc,
               corrpriors = list(phi = phiprior, omg = omgprior),
               corrfcn = corrf, kappa = kappa,
               corrtuning = list(phi = phisc, omg = omgsc, kappa = 0),
               dispersion = 1, test = 10)

### Full run
emc <- update(emt, test = FALSE)

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

## End(Not run)
```

`mcsglmm_mala` *MCMC samples from the Spatial GLMM*

Description

Draw MCMC samples from the Spatial GLMM with known link function

Usage

```
mcsglmm_mala(
  formula,
  family = "gaussian",
  data,
  weights,
  subset,
  offset,
  atsample,
  corrfcn = "matern",
  linkp,
  phi,
  omg,
  kappa,
  Nout,
  Nthin = 1,
  Nbi = 0,
  betm0,
  betQ0,
```

```
  ssqdf,
  ssqsc,
  corrpriors,
  corrtuning,
  malatuning,
  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.
offset	See lm.
atsample	A formula in the form ~ x1 + x2 + ... + xd with the coordinates of the sampled locations.
corrfcn	Spatial correlation function. See geoBayes_correlation for details.
linkp	Parameter of the link function. A scalar value.
phi	Optional starting value for the MCMC for the spatial range parameter phi. Defaults to the mean of its prior. If corrtuning[["phi"]] is 0, then this argument is required and it corresponds to the fixed value of phi. This can be a vector of the same length as Nout.
omg	Optional starting value for the MCMC for the relative nugget parameter omg. Defaults to the mean of its prior. If corrtuning[["omg"]] 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 Nout.
kappa	Optional starting value for the MCMC for the spatial correlation parameter kappa (Matern smoothness or exponential power). Defaults to the mean of its prior. If corrtuning[["kappa"]] is 0 and it is needed for the chosen correlation function, then this argument is required and it corresponds to the fixed value of kappa. This can be a vector of the same length as Nout.
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.
corrpriors	A list with the components phi, omg and kappa as needed. These correspond to the prior distribution parameters. For phi and omg it must be a vector of length 4. The generalized inverse gamma prior is assumed and the input corresponds to the parameters scale, shape, exponent, location in that order (see Details). For kappa it must be a vector of length 2. A uniform prior is assumed and the input corresponds to the lower and upper bounds in that order.
corrtuning	A vector or list with the components phi, omg and kappa as needed. These correspond to the random walk parameter for the Metropolis-Hastings step. Smaller values increase the acceptance ratio. Set this to 0 for fixed parameter value.
malatuning	Tuning parameter for the MALA updates.
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 phi and omg. If set to TRUE, the acceptance ratio will be printed on the screen every 100 iterations of the MCMC. Tune the phisc and omgsc 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 phi is defined by

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

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

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

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

Value

A list containing the objects MODEL, DATA, FIXED, MCMC and call. The MCMC samples are stored in the object MCMC 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
- phi A vector with the MCMC samples for the spatial range parameter, if sampled.
- omg A vector with the MCMC samples for the relative nugget parameter, if sampled.
- 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, if sampled.
- sys_time The total computing time for the MCMC sampling.
- Nout, Nbi, Nthin As in input. Used internally in other functions.

The other objects contain input variables. The object call contains the function call.

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
corrf <- "spherical"
family <- "binomial.probit"
kappa <- 0
ssqdf <- 1
ssqsc <- 1
betm0 <- 0
betQ0 <- .01
phiprior <- c(100, 1, 1000, 100) # U(100, 200)
phisc <- 3
omgprior <- c(2, 1, 1, 0)        # Exp(mean = 2)
omgsc <- .1
##'
### MCMC sizes
Nout <- 100
Nthin <- 1
Nbi <- 0

### Trial run
emt <- mcsglmm_mala(Infected ~ 1, family, rhizdata, weights = Total,
```

```
                atsample = ~ Xcoord + Ycoord,
                Nout = Nout, Nthin = Nthin, Nbi = Nbi,
                betm0 = betm0, betQ0 = betQ0, ssqdf = ssqdf, ssqsc = ssqsc,
                corrpriors = list(phi = phiprior, omg = omgprior),
                corrfcn = corrf, kappa = kappa,
                corrtuning = list(phi = phisc, omg = omgsc, kappa = 0),
                malatuning = .003, dispersion = 1, test = 10)

### Full run
emc <- update(emt, test = FALSE)

emcmc <- mcmcmake(emc)
summary(emcmc[, c("phi", "omg", "beta", "ssq")])
plot(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,
  offset,
  atsample,
  corrfcn = "matern",
  linkp,
  phi,
  omg,
  kappa,
  Nout,
  Nthin = 1,
  Nbi = 0,
  betm0,
  betQ0,
  ssqdf,
  ssqsc,
  tsqdf,
  tsqsc,
  corrpriors,
```

```
  corrtuning,
  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.
offset	See lm.
atsample	A formula in the form ~ x1 + x2 + ... + xd with the coordinates of the sampled locations.
corrfcn	Spatial correlation function. See geoBayes_correlation for details.
linkp	Parameter of the link function. A scalar value.
phi	Optional starting value for the MCMC for the spatial range parameter phi. Defaults to the mean of its prior. If corrtuning[["phi"]] is 0, then this argument is required and it corresponds to the fixed value of phi. This can be a vector of the same length as Nout.
omg	Optional starting value for the MCMC for the relative nugget parameter omg. Defaults to the mean of its prior. If corrtuning[["omg"]] 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 Nout.
kappa	Optional starting value for the MCMC for the spatial correlation parameter kappa (Matern smoothness or exponential power). Defaults to the mean of its prior. If corrtuning[["kappa"]] is 0 and it is needed for the chosen correlation function, then this argument is required and it corresponds to the fixed value of kappa. This can be a vector of the same length as Nout.
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.
corrpriors	A list with the components phi, omg and kappa as needed. These correspond to the prior distribution parameters. For phi and omg it must be a vector of length 4. The generalized inverse gamma prior is assumed and the input corresponds to the parameters scale, shape, exponent, location in that order (see Details). For kappa it must be a vector of length 2. A uniform prior is assumed and the input corresponds to the lower and upper bounds in that order.
corrtuning	A vector or list with the components phi, omg and kappa as needed. These correspond to the random walk parameter for the Metropolis-Hastings step. Smaller values increase the acceptance ratio. Set this to 0 for fixed parameter value.
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 phi and omg. If set to TRUE, the acceptance ratio will be printed on the screen every 100 iterations of the MCMC. Tune the phisc and omgsc 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

Simulates from the posterior distribution of this model.

Value

A list containing the objects MODEL, DATA, FIXED, MCMC and call. The MCMC samples are stored in the object MCMC 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, if sampled.
- omg A vector with the MCMC samples for the relative nugget parameter, if sampled.
- 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, if sampled.
- sys_time The total computing time for the MCMC sampling.

- Nout, Nbi, Nthin As in input. Used internally in other functions.

The other objects contain input variables. The object call contains the function call.

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
linkp <- 1

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

samplt <- 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,
                  corrprior = list(phi = phiprior, omg = omgprior),
                  linkp = linkp,
                  corrtuning = list(phi = phisc, omg = omgsc, kappa = 0),
                  test=10)

sample <- update(samplt, test = FALSE)

## End(Not run)
```

---

mcstrga_mala *MCMC samples from the transformed Gaussian model*

Description

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

Usage

```
mcstrga_mala(
  formula,
  data,
  weights,
  subset,
  offset,
  atsample,
  corrfcn = "matern",
  linkp,
  phi,
  omg,
  kappa,
  Nout,
  Nthin = 1,
  Nbi = 0,
  betm0,
  betQ0,
  ssqdf,
  ssqsc,
  tsqdf,
  tsqsc,
  corrpriors,
  corrtuning,
  malatuning,
  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.
offset	See `lm`.
atsample	A formula in the form `~ x1 + x2 + ... + xd` with the coordinates of the sampled locations.

corrfcn	Spatial correlation function. See geoBayes_correlation for details.
linkp	Parameter of the link function. A scalar value.
phi	Optional starting value for the MCMC for the spatial range parameter phi. Defaults to the mean of its prior. If corrtuning[["phi"]] is 0, then this argument is required and it corresponds to the fixed value of phi. This can be a vector of the same length as Nout.
omg	Optional starting value for the MCMC for the relative nugget parameter omg. Defaults to the mean of its prior. If corrtuning[["omg"]] 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 Nout.
kappa	Optional starting value for the MCMC for the spatial correlation parameter kappa (Matern smoothness or exponential power). Defaults to the mean of its prior. If corrtuning[["kappa"]] is 0 and it is needed for the chosen correlation function, then this argument is required and it corresponds to the fixed value of kappa. This can be a vector of the same length as Nout.
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.
corrpriors	A list with the components phi, omg and kappa as needed. These correspond to the prior distribution parameters. For phi and omg it must be a vector of length 4. The generalized inverse gamma prior is assumed and the input corresponds to the parameters scale, shape, exponent, location in that order (see Details). For kappa it must be a vector of length 2. A uniform prior is assumed and the input corresponds to the lower and upper bounds in that order.
corrtuning	A vector or list with the components phi, omg and kappa as needed. These correspond to the random walk parameter for the Metropolis-Hastings step. Smaller values increase the acceptance ratio. Set this to 0 for fixed parameter value.
malatuning	Tuning parameter for the MALA updates.
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 `phi` and `omg`. If set to `TRUE`, the acceptance ratio will be printed on the screen every 100 iterations of the MCMC. Tune the `phisc` and `omgsc` 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

Simulates from the posterior distribution of this model.

Value

A list containing the objects `MODEL`, `DATA`, `FIXED`, `MCMC` and `call`. The MCMC samples are stored in the object `MCMC` 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, if sampled.
- `omg` A vector with the MCMC samples for the relative nugget parameter, if sampled.
- `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`, if sampled.
- `sys_time` The total computing time for the MCMC sampling.
- `Nout`, `Nbi`, `Nthin` As in input. Used internally in other functions.

The other objects contain input variables. The object `call` contains the function call.

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
linkp <- 1

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

samplt <- mcstrga_mala(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,
                  corrprior = list(phi = phiprior, omg = omgprior),
                  linkp = linkp,
                  corrtuning = list(phi = phisc, omg = omgsc, kappa = 0),
                  malatuning = .0002, test=10)

sample <- update(samplt, 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.
- inxygrid A logical vector indicating which rows of xycoord fall inside borders, and therefore correspond to the 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

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

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
corrf <- "spherical"
kappa <- 0
ssqdf <- 1
ssqsc <- 1
betm0 <- 0
betQ0 <- .01
family <- "binomial.probit"
```

```
### Skeleton points
philist <- c(100, 140, 180)
omglist <- c(.5, 1)
parlist <- expand.grid(linkp=0, phi=philist, 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, family, rhizoctonia, weights = Total,
                       atsample = ~ Xcoord + Ycoord,
                       Nout = Nout, Nthin = Nthin, Nbi = Nbi,
                       betm0 = betm0, betQ0 = betQ0,
                       ssqdf = ssqdf, ssqsc = ssqsc,
                       phi = parlist$phi[i], omg = parlist$omg[i],
                       linkp = parlist$linkp[i], kappa = parlist$kappa[i],
                       corrfcn = corrf,
                       corrtuning=list(phi = 0, omg = 0, kappa = 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` *Reverse logistic regression estimation*

Description

Perform the reverse logistic regression estimation

Usage

```
revlogreg(lglk, N)
```

Arguments

`lglk`	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(lglk) or a scalar corresponding to the sample sizes from each model. Must sum(N) == nrow(lglk). 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.

References

Geyer, C. J. (1994). Estimating normalizing constants and reweighting mixtures in Markov chain Monte Carlo. Technical Report 568, School of Statistics, University of Minnesota.

---

rhizoctonia *Rhizoctonia root rot infections*

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

See reference below.

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,
  offset,
  atsample,
  beta,
  linkp,
  phi,
  omg,
  kappa,
  ssq,
  corrfcn = "matern",
  longlat = FALSE,
  dispersion = 1,
  returnGRF = FALSE,
  warndisp = TRUE
)

rstrga(
  n,
  formula,
  data,
  weights,
  subset,
  offset,
  atsample,
  beta,
  linkp,
  phi,
```

```
  omg,
  kappa,
  ssq,
  corrfcn = "matern",
  longlat = FALSE,
  dispersion = 1,
  returnGRF = FALSE
)

rsgrf(
  n,
  formula,
  data,
  subset,
  offset,
  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.
offset	See `lm`.
atsample	A formula of the form `~ Xcoord + Ycoord` specifying the sampled locations.
beta	A vector of the regressor coefficents 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, no such distribution exists so an approximate sample is returned. Use with caution.
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 = l,
                 atsample = ~ Xcoord + Ycoord, beta = beta, linkp = linkp,
                 phi = phi, omg = omg, kappa = kappa, ssq = ssq,
                 corrfcn = corrf, returnGRF = TRUE)

## End(Not run)
```

select_proposals *Selection of multiple importance sampling distributions*

Description

Selection of multiple importance sampling distributions

Usage

```
select_proposals_SEQ(
  pargrid,
  K,
  istart,
  relativeSE = FALSE,
  N1,
  N2,
  Nthin,
  Nbi,
  formula,
  family = "gaussian",
  data,
  weights,
  subset,
  offset,
  atsample,
  corrfcn = "matern",
  betm0,
  betQ0,
  ssqdf,
  ssqsc,
  dispersion = 1,
  longlat = FALSE,
  nbatch1 = 0.5,
  nbatch2 = 0.5,
  S1method = c("RL", "MW"),
  bvmethod = c("Standard", "TukeyHanning", "Bartlett"),
  transf = c("no", "mu", "wo")
)

select_proposals_MNX(
  pargrid,
  istart,
  nfix,
  relativeSE = FALSE,
  N1,
  N2,
  Nthin,
```

```
  Nbi,
  cooling,
  formula,
  family = "gaussian",
  data,
  weights,
  subset,
  offset,
  atsample,
  corrfcn = "matern",
  betm0,
  betQ0,
  ssqdf,
  ssqsc,
  dispersion = 1,
  longlat = FALSE,
  nbatch1 = 0.5,
  nbatch2 = 0.5,
  S1method = c("RL", "MW"),
  bvmethod = c("Standard", "TukeyHanning", "Bartlett"),
  transf = c("no", "mu", "wo"),
  verbose = FALSE
)

select_proposals_ENT(
  pargrid,
  istart,
  nfix,
  relativeSE = FALSE,
  N1,
  Nthin,
  Nbi,
  cooling,
  formula,
  family = "gaussian",
  data,
  weights,
  subset,
  offset,
  atsample,
  corrfcn = "matern",
  betm0,
  betQ0,
  ssqdf,
  ssqsc,
  dispersion = 1,
  longlat = FALSE,
  nbatch1 = 0.5,
```

```
  nbatch2 = 0.5,
  S1method = c("RL", "MW"),
  bvmethod = c("Standard", "TukeyHanning", "Bartlett"),
  transf = c("no", "mu", "wo"),
  verbose = FALSE
)
```

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.
K	How many proposal densities in total to choose among the rows of pargrid. Needed for SEQ only. For MNX and ENT this is determined by the length of istart.
istart	Start with these rows of pargrid. A vector of indices.
relativeSE	Logical. Whether the choice is based on the standard error (FALSE), or relative standard error (TRUE).
N1	The sample size for stage 1.
N2	The sample sie for stage 2.
Nthin	Thinning
Nbi	Burn-in
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.
offset	See lm.
atsample	A formula in the form ~ x1 + x2 + ... + xd with the coordinates of the sampled locations.
corrfcn	Spatial correlation function. See geoBayes_correlation for details.
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.
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.
S1method	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.
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.
nfix	In the case of MNX and ENT, the first nfix elements of istart will always be included.
cooling	A decreasing sequence of temperature values for the simulated annealing. All elements must be positive. A suggested value is Tinit / log(((0:N) %/% Tstp)*Tstp + exp(1)) for N+1 iterations, where Tinit is the initial temperature and Tstp is the number of iterations before the temperature is reduced.
verbose	Logical. Prints information about the simulated annealing.

Details

SEQ is a sequential method starting with istart and additng to it until K proposals have been selected. At each iteration, the point with the highest (relative?) standard error is added

MNX is the minimax method. The chosen proposal corresponds to the lowest maximum (relative?) standard error.

ENT is the entropy method. The chosen proposal corresponds to the highest determinant of the (relative?) covariance matrix at the first stage.

Value

A list with components

selected The rows of pargrid selected.

isel The indices of the rows of pargrid selected.

se The standard error corresponding to the selected parameters.

samples A list containing the samples from the selected parameters.

References

Roy, V., & Evangelou, E. (2018). Selection of proposal distributions for generalized importance sampling estimators. arXiv preprint arXiv:1805.00829.

Examples

```
## Not run:
data(rhizoctonia)
### Define the model
corrf <- "spherical"
kappa <- 0
ssqdf <- 1
ssqsc <- 1
betm0 <- 0
betQ0 <- .01
family <- "binomial.probit"
formula <- Infected ~ 1
atsample <- ~ Xcoord + Ycoord
### Skeleton points
philist <- seq(100, 200, 10)
omglist <- 0
parlist <- expand.grid(linkp=0, phi=philist, omg=omglist, kappa = kappa)
### MCMC sizes
Nout <- 100
Nthin <- 1
Nbi <- 10
## Select proposals
K <- 3                          # Choose 3 proposals
istart_SEQ <- 6                 # Start with middle
istart_MNX <- istart_ENT <- c(6, 2, 10)
cooling_MNX <- .05/log((0:24 %/% 5)*5 + exp(1))
cooling_ENT <- .3/log((0:49 %/% 10)*10 + exp(1))
prop_SEQ <- select_proposals_SEQ(pargrid = parlist, K = K,
                                 istart = istart_SEQ,
                                 relativeSE = TRUE,
                                 N1 = Nout, N2 = Nout,
                                 Nthin = Nthin, Nbi = Nbi,
                                 formula = formula, family = family,
                                 data = rhizoctonia, weights = Total,
                                 atsample = atsample, corrfcn = corrf,
                                 betm0 = betm0, betQ0 = betQ0,
                                 ssqdf = ssqdf, ssqsc = ssqsc,
                                 dispersion = 1, longlat = FALSE,
                                 nbatch1 = 0.5, nbatch2 = 0.5,
                                 bvmethod = "TukeyHanning",
                                 transf = "mu")
prop_MNX <- select_proposals_MNX(pargrid = parlist,
                                 istart = istart_MNX, nfix = 1L,
                                 cooling = cooling_MNX,
                                 relativeSE = TRUE,
                                 N1 = Nout, N2 = Nout,
                                 Nthin = Nthin, Nbi = Nbi,
```

```
                                formula = formula, family = family,
                                data = rhizoctonia, weights = Total,
                                atsample = atsample, corrfcn = corrf,
                                betm0 = betm0, betQ0 = betQ0,
                                ssqdf = ssqdf, ssqsc = ssqsc,
                                dispersion = 1, longlat = FALSE,
                                nbatch1 = 0.5, nbatch2 = 0.5,
                                bvmethod = "TukeyHanning",
                                transf = "mu",
                                verbose = TRUE)
prop_ENT <- select_proposals_ENT(pargrid = parlist,
                                istart = istart_ENT, nfix = 1L,
                                cooling = cooling_ENT,
                                relativeSE = TRUE,
                                N1 = Nout,
                                Nthin = Nthin, Nbi = Nbi,
                                formula = formula, family = family,
                                data = rhizoctonia, weights = Total,
                                atsample = atsample, corrfcn = corrf,
                                betm0 = betm0, betQ0 = betQ0,
                                ssqdf = ssqdf, ssqsc = ssqsc,
                                dispersion = 1, longlat = FALSE,
                                nbatch1 = 0.5, nbatch2 = 0.5,
                                bvmethod = "TukeyHanning",
                                transf = "mu",
                                verbose = TRUE)

## End(Not run)
```

`spcovariance` *Spatial variance-covariance matrix*

Description

Calculates the spatial variance-covariance matrix for a selection of correlation functions.

Usage

```
spcovariance(...)

## S3 method for class 'formula'
spcovariance(
  formula,
  data,
  subset,
  corrfcn,
  ssq,
  phi,
  omg,
```

```
  kappa,
  longlat = FALSE,
  ...
)

## S3 method for class 'numeric'
spcovariance(x, corrfcn, ssq, phi, omg, kappa, ...)

## S3 method for class 'dist'
spcovariance(x, corrfcn, ssq, phi, omg, kappa, ...)
```

Arguments

...	Further arguments. Not currently in use.
formula	A formula of the form ~ Xcoord + Ycoord specifying the sampled locations.
data	An optional data frame containing the variables in the model.
subset	An optional set of indices. The covariance will be calculated for those coordinates only.
corrfcn	The correlation function to use.
ssq	The partial sill parameter.
phi	The spatial range parameter.
omg	The relative nugget parameter.
kappa	The spatial smoothness parameter.
longlat	How to compute the distance between locations. If FALSE, Euclidean distance, if TRUE Great Circle distance. See spDists.
x	A numerical object of distances.

Value

For a formula input, a variance-covariance matrix. For a numeric input, an object of the the same dimensions as its first input.

---

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 mcsglmm or mcstrga.
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 joint log likelihood log f(y,T(z)|parameters) where T(z) is the transformation, for each (y,z) in runs and for parameters in pargrid up to a constant which does not depend on the parameters. The parameters beta and sigma^2 are integrated out.

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.

---

sploglik_cross *Spatial log likelihood*

Description

Spatial joint log likelihood

Usage

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

Arguments

runs	A list with outputs from the function mcsglmm or mcstrga.
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. The input can also be a vector (of the same length as runs to allow for different transformation to be used when evaluating each likelihood.

Details

Computes the joint log likelihood log f(y,T(z)|parameters) where T(z) is the transformation, for each (y,z) in runs and for parameters in runs up to a constant which does not depend on the parameters. The parameters beta and sigma^2 are integrated out.

Value

A matrix with number of rows the total number of samples in runs and number of columns the length of runs. 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 *Combine* data.frames

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 class of each variable. The elements in fillwith 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` *Subset MCMC chain*

Description

Return subset of MCMC chain.

Usage

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

Arguments

`x`	Output from the functions `mcsglmm` or `mcstrga`.
`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.

Index