Package ‘midasINLA’

September 17, 2026

Title Spatial MIDAS Models Using INLA

Version 0.1.0

Description Provides tools for fitting spatial Mixed Data Sampling (MIDAS) regression models using Integrated Nested Laplace Approximation (INLA). The package is designed for settings where responses and explanatory variables are observed at different temporal frequencies and supports both constant and spatially varying regression coefficients.

License GPL-3

Encoding UTF-8

RoxygenNote 8.0.0

Imports matrixStats, Matrix, stats

Suggests INLA, knitr, rmarkdown, testthat (>= 3.0.0), dplyr, ggplot2, tidyr

Additional_repositories https://inla.r-inla-download.org/R/stable

VignetteBuilder knitr

Depends R (>= 4.2)

LazyData true

Config/testthat/edition 3

NeedsCompilation no

Author Stephen Jun Villejo [aut, cre]

Maintainer Stephen Jun Villejo <s.villejo@imperial.ac.uk>

Repository CRAN

Date/Publication 2026-09-17 12:30:02 UTC

Contents

`compute_beta_spatial` *Compute posterior summaries of MIDAS coefficients*

Description

Computes posterior summaries of the MIDAS regression coefficients from a fitted spatial MIDAS model returned by `fit_Minla_spatial()`. The output depends on whether the MIDAS coefficient is spatially varying and on the specified spatial prior.

Usage

```
compute_beta_spatial(model, n_loc)
```

Arguments

`model`	A fitted spatial MIDAS model returned by `fit_Minla_spatial()`.
`n_loc`	Integer specifying the number of spatial locations for which coefficient summaries should be computed.

Details

For spatially varying coefficients with an ICAR prior, the function returns summaries and marginal distributions for the global coefficient, the location-specific spatial deviations, and the resulting location-specific total coefficients. Posterior samples of the total coefficients are also returned.

For spatially varying coefficients with an IID prior, the function returns posterior summaries and marginal distributions for the location-specific coefficients.

For non-spatially varying coefficients, the function returns the posterior marginal distribution and summary of the MIDAS coefficient associated with each high-frequency covariate.

Value

A list containing one element for each high-frequency MIDAS covariate in `model$hf_input`. The elements are named `"hf_index_1"`, `"hf_index_2"`, and so on. The contents of each element depend on the spatial structure of the corresponding MIDAS covariate:

Spatially varying coefficient with ICAR prior A list containing `summary.global.beta`, `marginal.global.beta`, `summary.icar.beta`, `marginal.icar.beta`, `summary.total.beta`, and `sample.total.beta`.

Spatially varying coefficient with IID prior A list containing `summary.beta` and `marginal.beta`.

Non-spatially varying coefficient A list containing `summary.beta` and `marginal.beta`.

The summary data frames contain the posterior mean, standard deviation, and 2.5%, 50%, and 97.5% posterior quantiles.

Examples

```
if (requireNamespace("INLA", quietly = TRUE)) {
  INLA::inla.setOption(num.threads = 1)

  data(data_spatialpoisson_example)

  # Read the spatial adjacency graph
  g <- INLA::inla.read.graph(
    filename = system.file("map.adj", package = "midasINLA")
  )

  # Prepare a spatial MIDAS predictor
  Midas_x1 <- prepare_Minla_spatial(
    x = data_spatialpoisson_example$data_x1$x1,
    loc_x = data_spatialpoisson_example$data_x1$loc,
    constraint = "hyperbolic",
    K = 0:29,
    m = 30,
    svc = TRUE,
    svc_prior = "icar",
    g = g
  )

  # Fit the spatial Poisson MIDAS model
  fit <- fit_Minla_spatial(
    formula = y ~ 1,
    data = data_spatialpoisson_example$data_y,
    loc_var = "loc",
    time_var = "Time",
    family = "poisson",
    hf_input = list(Midas_x1),
    inla_options = list(
      verbose = FALSE,
      control.predictor = list(
        compute = TRUE,
        link = 1
      )
    )
  )

  # Compute posterior summaries of the MIDAS coefficients
  beta_summary <- compute_beta_spatial(
    model = fit,
    n_loc = 16
  )

  # Inspect summaries of the location-specific total coefficients
  beta_summary$hf_index_1$summary.total.beta
}
```

compute_weights *Compute posterior estimates of MIDAS lag weights*

Description

Computes posterior estimates of the normalized MIDAS lag weights from a fitted MIDAS model returned by `fit_Minla_spatial()`. The function draws samples from the posterior marginal distributions of the MIDAS hyperparameters and uses these samples to obtain the corresponding lag-weight functions.

Usage

```
compute_weights(model, n.samples = 200)
```

Arguments

`model`	A fitted MIDAS model returned by `fit_Minla_spatial()`.
`n.samples`	Positive integer specifying the number of posterior samples drawn from the MIDAS hyperparameter marginal distributions to estimate the lag-weight distribution. Defaults to `200`.

Details

The supported MIDAS lag constraints are `"hyperbolic"`, `"gaussian"`, `"beta1"`, `"beta2"`, and `"almon2"`. For each high-frequency covariate, the resulting weights are normalized to sum to one across all included lags.

Value

A list containing one data frame for each high-frequency covariate in `model$hf_input`. The elements are named `"hf_1"`, `"hf_2"`, and so on. Each data frame contains:

lag The MIDAS lag index.

mean The posterior mean of the normalized lag weight.

q2.5 The 2.5% posterior quantile of the lag weight.

q97.5 The 97.5% posterior quantile of the lag weight.

Examples

```
if (requireNamespace("INLA", quietly = TRUE)) {
  data(data_spatialpoisson_example)

  g <- INLA::inla.read.graph(
    filename = system.file("map.adj", package = "midasINLA")
  )

  Midas_x1 <- prepare_Minla_spatial(
```

```
    x = data_spatialpoisson_example$data_x1$x1,
    loc_x = data_spatialpoisson_example$data_x1$loc,
    constraint = "hyperbolic",
    K = 0:29,
    m = 30,
    svc = TRUE,
    svc_prior = "icar",
    g = g
  )

  fit <- fit_Minla_spatial(
    formula = y ~ 1,
    data = data_spatialpoisson_example$data_y,
    loc_var = "loc",
    time_var = "Time",
    family = "poisson",
    hf_input = list(Midas_x1),
    inla_options = list(
      verbose = FALSE,
      control.predictor = list(
        compute = TRUE,
        link = 1
      )
    )
  )

  weights <- compute_weights(
    model = fit,
    n.samples = 200
  )

  head(weights$hf_1)
}
```

---

data_binomial_example *Example MIDAS dataset*

Description

This dataset contains simulated high-frequency covariates and a binomial response constructed using MIDAS lag weights.

Usage

```
data_binomial_example
```

Format

A list with the following components:

x1 Numeric vector of high-frequency covariate 1

x2 Numeric vector of high-frequency covariate 2

y Integer vector of binomial response counts

p Underlying success probabilities

Ntrials Number of trials for each observation

weights1 Lag weights used in the MIDAS structure for covariate 1

weights2 Lag weights used in the MIDAS structure for covariate 2

beta0 True value of intercept β_0

beta1 True value of β_1

beta2 True value of β_2

beta3 True value of β_3

Details

Simulated dataset generated using a binomial sampling model

The dataset is generated using two covariates. The first one has a hyperbolic weighting scheme with parameter gamma = 0.9 and lag length 13. The second one has gaussian weighting scheme with parameters mu = 8, sigma = 7, and lag length of 20.

Source

Simulated data

`data_spatialpoisson_example`

Example MIDAS dataset

Description

This dataset contains simulated high-frequency covariates and a poisson response constructed using MIDAS lag weights.

Usage

```
data_spatialpoisson_example
```

Format

A list with the following components:

data_x1 High-frequency covariate 1 data frame

data_x2 High-frequency covariate 2 data frame

data_y Response data frame

weights1 Lag weights used in the MIDAS structure for covariate 1

weights2 Lag weights used in the MIDAS structure for covariate 2

eta Linear predictor log lambda

beta0 True value of intercept β_0

beta1 True value of β_1

beta2 True value of β_2

icar Vector of region-specific deviations from β_1

tau Precision parameter of the ICAR model

Details

Simulated dataset generated using a spatial poisson sampling model

The dataset is generated using two covariates. The first one has a hyperbolic weighting scheme with parameter gamma = 0.9 and lag length 29. The second one has Gaussian constraint with parameters mu = 10 and sigma = 12.

Source

Simulated data

`fit_Minla_spatial` *Fit a spatial MIDAS model using INLA*

Description

Fits a Mixed Data Sampling (MIDAS) regression model with optional spatially varying coefficients using Integrated Nested Laplace Approximation (INLA). High-frequency covariates are supplied as MIDAS objects created by `prepare_Minla_spatial()`.

Usage

```
fit_Minla_spatial(
  formula,
  data,
  loc_var,
  time_var,
  family,
  hf_input = NULL,
  Ntrials = NULL,
  E = NULL,
  inla_options = list()
)
```

Arguments

formula	A model formula specifying the response and other covariates. The response variable must be a column in data.
data	A data frame containing the response and any additional model covariates. It must contain the variables specified by loc_var and time_var.
loc_var	Character string specifying the name of the location variable in data.
time_var	Character string specifying the name of the time variable in data.
family	Character string specifying the likelihood family for the response. For example, "poisson" or "binomial".
hf_input	A list of MIDAS objects returned by prepare_Minla_spatial(). Each object specifies a high-frequency covariate and its MIDAS lag structure. Multiple MIDAS objects can be supplied.
Ntrials	Optional vector specifying the number of trials for a binomial response. Used only when family = "binomial".
E	Optional vector of expected counts or exposure values for a Poisson model. Its length must match the number of rows in data. Used only when family = "poisson".
inla_options	A named list of additional arguments passed to INLA::inla(). By default, control.compute$config is set to TRUE if it is not already specified.

Details

The function constructs the INLA model by incorporating the MIDAS components specified in hf_input. Multiple high-frequency covariates can be included by supplying multiple MIDAS objects in hf_input.

Value

A list containing:

formula_final The final INLA model formula, including the MIDAS components.

data_final The response data used for model fitting after ordering by location and time and removing rows with incomplete lagged covariate information.

rm_max The maximum number of initial observations removed across locations because of incomplete MIDAS lagged covariates.

res The fitted INLA model returned by INLA::inla().

hf_input The list of MIDAS objects supplied through hf_input.

Examples

```
if (requireNamespace("INLA", quietly = TRUE)) {
  data(data_spatialpoisson_example)

  # Read the spatial adjacency graph
  g <- INLA::inla.read.graph(
    filename = system.file("map.adj", package = "midasINLA")
```

```
  )

  # Prepare a spatial MIDAS predictor
  Midas_x1 <- prepare_Minla_spatial(
    x = data_spatialpoisson_example$data_x1$x1,
    loc_x = data_spatialpoisson_example$data_x1$loc,
    constraint = "hyperbolic",
    K = 0:29,
    m = 30,
    svc = TRUE,
    svc_prior = "icar",
    g = g
  )

  # Fit the spatial Poisson MIDAS model
  fit <- fit_Minla_spatial(
    formula = y ~ 1,
    data = data_spatialpoisson_example$data_y,
    loc_var = "loc",
    time_var = "Time",
    family = "poisson",
    hf_input = list(Midas_x1),
    inla_options = list(
      verbose = FALSE,
      control.predictor = list(
        compute = TRUE,
        link = 1
      )
    )
  )

  # Inspect the fitted INLA model
  fit$res
}
```

`predict_midas` *Generate posterior predictive samples from a MIDAS model*

Description

Generates posterior predictive samples from a fitted MIDAS model using posterior samples of the latent linear predictor. The function supports Gaussian, Poisson, and binomial response distributions.

Usage

```
predict_midas(model, family = "gaussian", Ntrials = NULL, nsamples = 1000)
```

Arguments

`model`	A fitted MIDAS model returned by `fit_Minla_spatial()`.
`family`	Character string specifying the likelihood family. Supported values are `"gaussian"`, `"poisson"`, and `"binomial"`. Defaults to `"gaussian"`.
`Ntrials`	Optional vector specifying the number of trials for each observation when `family = "binomial"`. If omitted, the function attempts to use the `Ntrials` column in `model$data_final`.
`nsamples`	Positive integer specifying the number of posterior samples used to generate the predictive distribution. Defaults to `1000`.

Details

For a Gaussian response, posterior samples of the observation variance are obtained from the posterior samples of the Gaussian precision parameter and used to generate predictive observations. For Poisson and binomial responses, observations are generated using the appropriate inverse-link function.

Value

A list with two components:

computed_y A list containing posterior predictive summaries:

- **mean** Posterior predictive mean for each observation.
- **sd** Posterior predictive standard deviation for each observation.
- **q2.5** 2.5% posterior predictive quantile for each observation.
- **q97.5** 97.5% posterior predictive quantile for each observation.

samples A list containing posterior samples of the latent predictor and, for Gaussian responses, the posterior samples of the observation variance.

Examples

```
if (requireNamespace("INLA", quietly = TRUE)) {
  data(data_spatialpoisson_example)

  Midas_x1 <- prepare_Minla_spatial(
    x = data_spatialpoisson_example$data_x1$x1,
    loc_x = data_spatialpoisson_example$data_x1$loc,
    constraint = "hyperbolic",
    K = 0:29,
    m = 30,
    svc = FALSE
  )

  fit <- fit_Minla_spatial(
    formula = y ~ 1,
    data = data_spatialpoisson_example$data_y,
    loc_var = "loc",
    time_var = "Time",
```

```
    family = "poisson",
    hf_input = list(Midas_x1),
    inla_options = list(verbose = FALSE)
  )

  predictions <- predict_midas(
    model = fit,
    family = "poisson",
    nsamples = 1000
  )

  # Inspect posterior predictive summaries
  head(predictions$computed_y$mean)
  head(predictions$computed_y$q2.5)
  head(predictions$computed_y$q97.5)
}
```

`prepare_Minla_spatial` *Prepare a spatial MIDAS object for INLA estimation*

Description

Constructs the MIDAS design matrix and associated model specifications for use with `fit_Minla_spatial()`. The function creates lagged high-frequency covariate values for each location and optionally specifies a spatially varying coefficient (SVC) component.

Usage

```
prepare_Minla_spatial(
  x,
  loc_x,
  constraint,
  K,
  m,
  svc = FALSE,
  svc_prior = "iid",
  g = NULL
)
```

Arguments

`x`	Numeric vector of high-frequency covariate observations.
`loc_x`	Vector identifying the location associated with each observation in `x`. The length of `loc_x` must equal the length of `x`.
`constraint`	Character string specifying the constraint used for the MIDAS lag-response association. Supported constraints include `"hyperbolic"`, `"gaussian"`, `"beta1"`, `"beta2"`, and `"almon2"`.
`K`	Numeric vector specifying the lags to be included in the MIDAS representation.

`m` Numeric vector specifying the number of high-frequency covariate observations associated with each response observation. A single value can be supplied when the number of observations is constant over time, or a vector can be supplied when this number varies across response times.

`svc` Logical; if `TRUE`, specifies a spatially varying coefficient model. Defaults to `FALSE`.

`svc_prior` Character string specifying the prior for the spatially varying coefficient component. Must be either `"icar"` or `"iid"`. Defaults to `"iid"`.

`g` An INLA graph object used for the `"icar"` prior. Required when `svc_prior = "icar"`.

Value

A list containing the MIDAS design matrix and model specifications. The returned object includes:

X_matrix The MIDAS design matrix, including a location index.

constraint The MIDAS constraint used for the lag-response association.

lag_k The maximum lag specified in `K`.

K The vector of lags used to construct the design matrix.

m The number of high-frequency observations associated with each response observation.

rm.row The number of initial rows removed from each location because of incomplete lagged observations.

svc Whether a spatially varying coefficient component is specified.

svc_prior The prior specified for the spatially varying coefficient component.

g The INLA graph object, included when `svc_prior = "icar"`.

Examples

```
if (requireNamespace("INLA", quietly = TRUE)) {
  data(data_spatialpoisson_example)

  # Prepare a MIDAS object using a hyperbolic lag constraint
  # and a spatially varying coefficient with an ICAR prior.
  g <- INLA::inla.read.graph(
    filename = system.file("map.adj", package = "midasINLA")
  )

  Midas_x1 <- prepare_Minla_spatial(
    x = data_spatialpoisson_example$data_x1$x1,
    loc_x = data_spatialpoisson_example$data_x1$loc,
    constraint = "hyperbolic",
    K = 0:29,
    m = 30,
    svc = TRUE,
    svc_prior = "icar",
    g = g
  )
```

```
  # Inspect the resulting MIDAS design matrix
  head(Midas_x1$X_matrix)
}
```

Index