Package: MRFtools (via r-universe)

March 13, 2025

Version 0.0-2

Date 2023-12-08

Title Tools for Constructing and Plotting Markov Random Fields in R for Graphical Data

Maintainer Eric J. Petersen <eric.j.pedersen@gmail.com>

Imports gratia, stats, sf, ape

Suggests testthat, vdiffr, mgcv, gamm4, sp

Description Utility functions for using Markov Random Field smooths in Generalized Additive Models fitted with the 'mgcv' package.

License GPL-3

LazyData true

URL https://github.com/eric-pedersen/MRFtools

BugReports https://github.com/eric-pedersen/MRFtools/issues

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Encoding UTF-8

Config/testthat/edition 3

Config/pak/sysreqs libgdal-dev gdal-bin libgeos-dev libicu-dev libssl-dev libproj-dev libsqlite3-dev libudunits2-dev

Repository https://gavinsimpson.r-universe.dev

RemoteUrl https://github.com/gavinsimpson/MRFtools

RemoteRef HEAD

RemoteSha c044f83a7e4f6be72045493c340478fa2022c362

Contents

`as.matrix.mrf_penalty` *Convert a MRF penalty object to a matrix*

Description

Convert a MRF penalty object to a matrix

Usage

```
## S3 method for class 'mrf_penalty'
as.matrix(x, ...)
```

Arguments

`x`	an object inheriting from class `"mrf_penalty"`
`...`	arguments passed to other methods

Examples

```
p <- mrf_penalty(1:10)
as.matrix(p)
```

get_config *Extract configuration details of an MRF penalty*

Description

Extract configuration details of an MRF penalty

Usage

```
get_config(penalty)
```

Arguments

penalty	an object of class "mrf_penalty"

Value

An object of class "mrf_config", a list.

get_labels *Extract MRF node labels from an MRF penalty*

Description

Extract MRF node labels from an MRF penalty

Usage

```
get_labels(penalty)
```

Arguments

penalty	an object of class "mrf_penalty"

get_mrf *Extract a fitted MRF*

Description

Extract a fitted MRF

Usage

```
get_mrf(object, ...)

## S3 method for class 'bam'
get_mrf(object, ...)

## S3 method for class 'gamm'
get_mrf(object, ...)

## S3 method for class 'list'
get_mrf(object, ...)

## S3 method for class 'gam'
get_mrf(object, term, ...)
```

Arguments

object	An object from which to extract the fitted MRF. Currently only for objects of classes gam, bam, and gamm, and GAMMs fitted by gamm4::gamm4().
...	Arguments passed to other methods.
term	character; the MRF term to extract. Can be a partial match to a term, which is matched against the smooth label.

Value

A object representing the fitted MRF

get_penalty *Extract a MRF penalty matrix*

Description

Extract a MRF penalty matrix

Usage

```
get_penalty(penalty, ...)

## S3 method for class 'mrf_penalty'
get_penalty(penalty, ...)
```

Arguments

penalty	an R object from which to extract the MRF penalty matrix.
...	arguments passed to other methods.

Value

A penalty matrix of class "matrix".

---

get_type *Extract the type of MRF from the penalty*

Description

Extract the type of MRF from the penalty

Usage

```
get_type(object)

## S3 method for class 'mrf_penalty'
get_type(object)

## S3 method for class 'mrf_config'
get_type(object)
```

Arguments

object	an object of class "mrf_penalty" or "mrf_config".

Value

A length 1 character vector containing the type of MRF penalty.

mrf_penalty *Markov Random Field Penalty*

Description

Markov Random Field Penalty

Usage

```
mrf_penalty(object, ...)
```

Arguments

object	an R object to create the MRF penalty from.
...	arguments passed to other methods.

mrf_penalty.dendrogram *MRF penalty from a dendrogram*

Description

MRF penalty from a dendrogram

Usage

```
## S3 method for class 'dendrogram'
mrf_penalty(object, node_labels = NULL, add_delta = FALSE, ...)
```

Arguments

object	an R object to create the MRF penalty from.
node_labels	character; a vector of alternative labels for the levels of the factor.
add_delta	numeric or logical; either the numeric value to add to the diagonal of the MRF penalty matrix, or a logical value indicating if such an adjustment should be made. The default is to not alter the diagonal of the penalty matrix.
...	arguments passed to other methods.

`mrf_penalty.factor` *Fully connected graph and random effect MRF penalties from a factor*

Description

Fully connected graph and random effect MRF penalties from a factor

Usage

```
## S3 method for class 'factor'
mrf_penalty(
  object,
  type = c("full", "individual"),
  node_labels = NULL,
  add_delta = FALSE,
  ...
)
```

Arguments

`object`	an R object to create the MRF penalty from.
`type`	character; one of `"full"` or `"individual"` indicating if a fully connected graph (`"full"`) or a random effect (random intercepts; `"individual"`) penalty is created.
`node_labels`	character; a vector of alternative labels for the levels of the factor.
`add_delta`	numeric or logical; either the numeric value to add to the diagonal of the MRF penalty matrix, or a logical value indicating if such an adjustment should be made. The default is to not alter the diagonal of the penalty matrix.
`...`	arguments passed to other methods.

Examples

```
# a factor
fv <- factor(letters[1:10])

# create the MRF penalty for a fully connected graph
p <- mrf_penalty(fv, type = "full")
p
as.matrix(p)

# create the MRF penalty equivalent of random effects
p <- mrf_penalty(fv, type = "individual")
p
as.matrix(p)
```

mrf_penalty.hclust *MRF penalty from a hclust object*

Description

MRF penalty from a hclust object

Usage

```
## S3 method for class 'hclust'
mrf_penalty(object, ...)
```

Arguments

object	an R object to create the MRF penalty from.
...	arguments passed to other methods.

mrf_penalty.numeric *First-order random walk MRF penalty from a numeric vector*

Description

First-order random walk MRF penalty from a numeric vector

Usage

```
## S3 method for class 'numeric'
mrf_penalty(
  object,
  type = c("linear", "cyclic"),
  node_labels = NULL,
  add_delta = FALSE,
  end_points = NULL,
  ...
)
```

Arguments

object	an R object to create the MRF penalty from.
type	character; one of "linear" or "cyclic" indicating if the observations form a cyclic series or not.
node_labels	character; a vector of alternative labels for the levels of the factor.
add_delta	numeric or logical; either the numeric value to add to the diagonal of the MRF penalty matrix, or a logical value indicating if such an adjustment should be made. The default is to not alter the diagonal of the penalty matrix.

end_points	numeric; an optional vector of length 2 providing the end points of the period of cycle.
...	arguments passed to other methods.

Examples

```
# linear
p <- mrf_penalty(1:10)
as.matrix(p)

# cyclic
p <- mrf_penalty(1:10, type = "cyclic")
as.matrix(p)

# cyclic with user end points
p <- mrf_penalty(1:10, type = "cyclic", end_points = c(0,11))
as.matrix(p)
```

---

mrf_penalty.phylo *MRF penalty from a phylogeny*

Description

MRF penalty from a phylogeny

Usage

```
## S3 method for class 'phylo'
mrf_penalty(object, node_labels = NULL, add_delta = FALSE, eps = 0, ...)
```

Arguments

object	an R object to create the MRF penalty from.
node_labels	character; a vector of alternative labels for the levels of the factor.
add_delta	numeric or logical; either the numeric value to add to the diagonal of the MRF penalty matrix, or a logical value indicating if such an adjustment should be made. The default is to not alter the diagonal of the penalty matrix.
eps	A value to add to the variance-covariance matrix diagonal to make it positive definite
...	arguments passed to other methods.

mrf_penalty.sf *MRF penalty from polygon or multi-polygon simple features*

Description

MRF penalty from polygon or multi-polygon simple features

Usage

```
## S3 method for class 'sf'
mrf_penalty(object, node_labels = NULL, buffer = NULL, add_delta = FALSE, ...)
```

Arguments

object	an R object to create the MRF penalty from.
node_labels	character; a vector of alternative labels for the levels of the factor.
buffer	numeric; buffer distance for all or for individual elements of the geometry. See argument dist in sf::st_buffer for details.
add_delta	numeric or logical; either the numeric value to add to the diagonal of the MRF penalty matrix, or a logical value indicating if such an adjustment should be made. The default is to not alter the diagonal of the penalty matrix.
...	arguments passed to other methods.

mrf_penalty.SpatialPolygons *MRF penalty from a SpatialPolygons*

Description

MRF penalty from a SpatialPolygons

Usage

```
## S3 method for class 'SpatialPolygons'
mrf_penalty(object, node_labels = NULL, buffer = NULL, add_delta = FALSE, ...)
```

Arguments

object	an R object to create the MRF penalty from.
node_labels	character; a vector of alternative labels for the levels of the factor.
buffer	numeric; buffer distance for all or for individual elements of the geometry. See argument dist in sf::st_buffer for details.
add_delta	numeric or logical; either the numeric value to add to the diagonal of the MRF penalty matrix, or a logical value indicating if such an adjustment should be made. The default is to not alter the diagonal of the penalty matrix.
...	arguments passed to other methods.

`mrf_penalty.SpatialPolygonsDataFrame` *MRF penalty from a SpatialPoylgonsDataFrame*

Description

MRF penalty from a SpatialPoylgonsDataFrame

Usage

```
## S3 method for class 'SpatialPolygonsDataFrame'
mrf_penalty(object, node_labels = NULL, buffer = NULL, add_delta = FALSE, ...)
```

Arguments

`object`	an R object to create the MRF penalty from.
`node_labels`	character; a vector of alternative labels for the levels of the factor.
`buffer`	numeric; buffer distance for all or for individual elements of the geometry. See argument `dist` in sf::st_buffer for details.
`add_delta`	numeric or logical; either the numeric value to add to the diagonal of the MRF penalty matrix, or a logical value indicating if such an adjustment should be made. The default is to not alter the diagonal of the penalty matrix.
`...`	arguments passed to other methods.

Index