

Package ‘SUMMER’

January 20, 2019

Type Package

Title Spatio-Temporal Under-Five Mortality Methods for Estimation

Version 0.2.2

Description Provides methods for estimating, projecting, and plotting spatio-temporal under-five mortality rates, described in Mercer et al. (2015) <doi:10.1214/15-AOAS872>.

URL <https://github.com/bryandmartin/SUMMER>

BugReports <https://github.com/bryandmartin/SUMMER/issues>

Depends R (>= 3.2)

License GPL (>= 2)

Imports maptools, survey, stats, spdep, survival, ggplot2, utils,
Matrix, reshape2

Encoding UTF-8

LazyData true

RoxygenNote 6.1.0

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

Suggests INLA, knitr, rmarkdown, readstata13, gridExtra, R.rsp

VignetteBuilder R.rsp, knitr

NeedsCompilation no

Author Bryan D Martin [cre, aut],
Zehang R Li [aut],
Yuan Hsiao [aut],
Jessica Godwin [aut],
Jon Wakefield [aut],
Samuel J Clark [aut]

Maintainer Bryan D Martin <bmartin6@uw.edu>

Repository CRAN

Date/Publication 2019-01-20 22:30:11 UTC

R topics documented:

| | |
|-------------------------------|-----------|
| SUMMER-package | 2 |
| aggregateSurvey | 3 |
| BRFSS | 4 |
| ChangeRegion | 4 |
| countrySummary | 5 |
| countrySummary_mult | 6 |
| DemoData | 7 |
| DemoData2 | 8 |
| DemoMap | 8 |
| DemoMap2 | 9 |
| expit | 9 |
| fitINLA | 10 |
| fitSpace | 12 |
| getBirths | 14 |
| KingCounty | 15 |
| logit | 15 |
| mapPlot | 16 |
| plot.projINLA | 17 |
| projINLA | 18 |
| read_shape | 20 |
| rst | 20 |
| simhyper | 22 |
| Index | 24 |

SUMMER-package

*SUMMER package documentation.***Description**

SUMMER provides methods for estimating, projecting, and plotting spatio-temporal under-five mortality rates.

Details

For details on the model implemented in this package, see Mercer et al. (2015) <doi:10.1214/15-AOAS872>.

The development version of the package will be maintained on <https://github.com/bryandmartin/SUMMER>.

| | |
|-----------------|---|
| aggregateSurvey | <i>Aggregate estimators from different surveys.</i> |
|-----------------|---|

Description

Aggregate estimators from different surveys.

Usage

```
aggregateSurvey(data)
```

Arguments

data Output from [countrySummary_mult](#)

Value

Estimators aggregated across surveys.

Examples

```
## Not run:
data(DemoData)
data(DemoMap)
years <- levels(DemoData[[1]]$time)

# obtain direct estimates
data <- countrySummary_mult(births = DemoData,
  years = years, idVar = "id",
  regionVar = "region", timeVar = "time",
  clusterVar = "~clustid+id",
  ageVar = "age", weightsVar = "weights",
  geo.recode = NULL)

# obtain maps
geo <- DemoMap$geo
mat <- DemoMap$Amat

# Simulate hyper priors
priors <- simhyper(R = 2, nsamp = 1e+05, nsamp.check = 5000, Amat = mat, only.iid = TRUE)

# combine data from multiple surveys
data <- aggregateSurvey(data)
utils::head(data)

## End(Not run)
```

 BRFSS

The BRFSS dataset

Description

The Behavioral Risk Factor Surveillance System (BRFSS) is an annual telephone health survey conducted by the Centers for Disease Control and Prevention (CDC) that tracks health conditions and risk behaviors in the United States and its territories since 1984. This BRFSS dataset contains 16124 observations. The 'diab2' variable is the binary indicator of Type II diabetes, 'strata' is the strata indicator and 'rwt_llcp' is the final design weight. Records with missing HRA code or diabetes status are removed from this dataset. See http://www.cdc.gov/brfss/annual_data/2013/pdf/Weighting_Data.pdf for more details of the weighting procedure.

Usage

BRFSS

Format

A data.frame of 26 variables.

 ChangeRegion

Function to map region names to a common set.

Description

Function to map region names to a common set.

Usage

```
ChangeRegion(data, Bmat, regionVar = "region")
```

Arguments

| | |
|-----------|---|
| data | Preprocessed data |
| Bmat | Matrix of changes. Each row corresponds to a region name possibly in the data files, and each column corresponds to a region after mapping. The values in the matrix are binary. The row names and column names need to be specified to the region names. |
| regionVar | String indicating the region variable. Defaults to 'region'. |

Value

Data after changing region names

Examples

```
# Construct a small test data
testdata <- data.frame(region = c("north", "south", "east",
  "south", "east"), index = c(1:5))

# Construct a changing rule: combining south and east
Bmat <- matrix(c(1, 0, 0, 0, 1, 1), 3, 2)
colnames(Bmat) <- c("north", "south and east")
rownames(Bmat) <- c("north", "south", "east")
print(Bmat)

# New data after transformation
test <- ChangeRegion(testdata, Bmat, "region")
print(test)
```

countrySummary

Obtain the Horvitz-Thompson direct estimates and standard errors using delta method for a single survey.

Description

Obtain the Horvitz-Thompson direct estimates and standard errors using delta method for a single survey.

Usage

```
countrySummary(births, years, idVar = "v002", regionVar = "region",
  timeVar = "per5", clusterVar = "~v001+v002", ageVar = "ageGrpD",
  weightsVar = "v005", geo.recode = NULL)
```

Arguments

| | |
|------------|--|
| births | A matrix child-month data from getBirths |
| years | String vector of the year intervals used |
| idVar | Variable name for ID, typically 'v002' |
| regionVar | Variable name for region, typically 'v024', for older surveys might be 'v101' |
| timeVar | Variable name for time, typically 'per5' |
| clusterVar | Variable name for cluster, typically '~v001 + v002' |
| ageVar | Variable name for age group, default assumes the variable is called 'ageGrpD' |
| weightsVar | Variable name for sampling weights, typically 'v005' |
| geo.recode | The recode matrix to be used if region name is not consistent across different surveys. See ChangeRegion . |

Value

a matrix of period-region summary of the Horvitz-Thompson direct estimates, the standard errors using delta method for a single survey, the 95% confidence interval, and the logit of the estimates.

See Also

[countrySummary_mult](#)

Examples

```
## Not run:
data(DemoData)
years <- c("85-89", "90-94", "95-99", "00-04", "05-09", "10-14")
u5m <- countrySummary(births = DemoData[[1]], years = years, idVar = "id",
  regionVar = "region", timeVar = "time", clusterVar = "~clustid+id",
  ageVar = "age", weightsVar = "weights", geo.recode = NULL)

## End(Not run)
```

| | |
|---------------------|--|
| countrySummary_mult | <i>Obtain the Horvitz-Thompson direct estimates and standard errors using delta method for multiple surveys.</i> |
|---------------------|--|

Description

Obtain the Horvitz-Thompson direct estimates and standard errors using delta method for multiple surveys.

Usage

```
countrySummary_mult(births, years, idVar = "v002",
  regionVar = "region", timeVar = "per5", clusterVar = "~v001+v002",
  ageVar = "ageGrpD", weightsVar = "v005", geo.recode = NULL)
```

Arguments

| | |
|------------|---|
| births | A list of child-month data from multiple surveys from getBirths . The name of the list is used as the identifier in the output. |
| years | String vector of the year intervals used |
| idVar | Variable name for ID, typically 'v002' |
| regionVar | Variable name for region, typically 'v024', for older surveys might be 'v101' |
| timeVar | Variable name for time, typically 'per5' |
| clusterVar | Variable name for cluster, typically '~v001 + v002' |
| ageVar | Variable name for age group, default assumes the variable is called 'ageGrpD' |
| weightsVar | Variable name for sampling weights, typically 'v005' |
| geo.recode | The recode matrix to be used if region name is not consistent across different surveys. See ChangeRegion . |

Value

a matrix of period-region summary of the Horvitz-Thompson direct estimates, the standard errors using delta method for a single survey, the 95% confidence interval, the logit of the estimates, and the survey labels.

See Also

[countrySummary](#)

Examples

```
## Not run:
data(DemoData)
years <- c("85-89", "90-94", "95-99", "00-04", "05-09", "10-14")
u5m <- countrySummary_mult(births = DemoData, years = years, idVar = "id",
  regionVar = "region", timeVar = "time", clusterVar = "~clustid+id",
  ageVar = "age", weightsVar = "weights", geo.recode = NULL)

## End(Not run)
```

DemoData

Fake dataset for vignette.

Description

A small fake dataset with 4 regions and 5 survey years. This does not represent any real country's data and are based on a subset of the model dataset provided by DHS.

Usage

```
DemoData
```

Format

A list of with five components, named by survey year.

Source

<https://dhsprogram.com/data/model-datasets.cfm>

DemoData2

Fake dataset for vignette.

Description

A small fake dataset with 8 regions and two response variables: age and tobacco.use. This does not represent any real country's data and are based on a subset of the model dataset provided by DHS.

Usage

DemoData2

Format

A data.frame of 7 variables.

Source

<https://dhsprogram.com/data/model-datasets.cfm>

DemoMap

Map dataset for vignette.

Description

Shapefiles are from 1995 Uganda Admin 1 regions provided by DHS, but the data do not represent real information about any country.

Usage

DemoMap

Format

An object of class list of length 2.

Details

- geo. Geographic map files
- Amat. Adjacency matrix for regions

Source

<https://spatialdata.dhsprogram.com/boundaries/#view=table&countryId=UG>

DemoMap2

Map dataset for vignette.

Description

Shapefiles are from 2014 Kenya Admin 1 regions provided by DHS, but the data do not represent real information about any country.

Usage

DemoMap2

Format

An object of class `list` of length 2.

Details

- `geo`. Geographic map files
- `Amat`. Adjacency matrix for regions

Source

<https://spatialdata.dhsprogram.com/boundaries/#view=table&countryId=KE>

expit

Expit transformation

Description

Expit transformation

Usage

`expit(x)`

Arguments

`x` data

Value

expit of `x`

Examples

```
x <- .5
expit(x)
```

fitINLA

Fit INLA models to direct estimators with a yearly model.

Description

Fit INLA models to direct estimators with a yearly model.

Usage

```
fitINLA(data, Amat, geo, formula = NULL, rw = 2, is.yearly = TRUE,
  year_names, year_range = c(1980, 2014), m = 5, na.rm = TRUE,
  redo.prior = FALSE, priors = NULL, type.st = 1, useHyper = FALSE,
  a.iid = NULL, b.iid = NULL, a.rw1 = NULL, b.rw1 = NULL,
  a.rw2 = NULL, b.rw2 = NULL, a.icar = NULL, b.icar = NULL)
```

Arguments

| | |
|------------|--|
| data | Combined dataset |
| Amat | Adjacency matrix for the regions |
| geo | Geo file |
| formula | INLA formula. Defaults to RW2, ICAR, IID time, IID, region, IID survey effect, IID time-region interaction, IID survey-region interaction, and IID survey-time-region interaction. |
| rw | Take values 1 or 2, indicating the order of random walk. |
| is.yearly | Logical indicator for fitting yearly or period model. |
| year_names | string vector of year names |
| year_range | Entire range of the years (inclusive) defined in year_names. |
| m | Number of years in each period. |
| na.rm | Logical indicator of whether to remove rows with NA values in the data. Default set to TRUE. |
| redo.prior | Logical indicator of whether to re-estimate hyperparameters |
| priors | priors from simhyper |
| type.st | type for space-time interaction |
| useHyper | option to manually set all hyperpriors |
| a.iid | hyperparameter for i.i.d random effects, only need if useHyper = TRUE |
| b.iid | hyperparameter for i.i.d random effects, only need if useHyper = TRUE |
| a.rw1 | hyperparameter for RW1 random effects, only need if useHyper = TRUE |

| | |
|--------|--|
| b.rw1 | hyperparameter for RW1 random effects, only need if useHyper = TRUE |
| a.rw2 | hyperparameter for RW2 random effects, only need if useHyper = TRUE |
| b.rw2 | hyperparameter for RW2 random effects, only need if useHyper = TRUE |
| a.icar | hyperparameter for ICAR random effects, only need if useHyper = TRUE |
| b.icar | hyperparameter for ICAR random effects, only need if useHyper = TRUE |

Value

INLA model fit using the provided formula, country summary data, and geographic data

See Also

[countrySummary](#)

Examples

```
## Not run:

data(DemoData)
data(DemoMap)
years <- levels(DemoData[[1]]$time)

# obtain direct estimates
data <- countrySummary_mult(births = DemoData,
  years = years, idVar = "id",
  regionVar = "region", timeVar = "time",
  clusterVar = "~clustid+id",
  ageVar = "age", weightsVar = "weights",
  geo.recode = NULL)

# obtain maps
geo <- DemoMap$geo
mat <- DemoMap$Amat

# Simulate hyperpriors
priors <- simhyper(R = 2, nsamp = 1e+05, nsamp.check = 5000, Amat = mat, only.iid = TRUE)

# combine data from multiple surveys
data <- aggregateSurvey(data)

# Model fitting with INLA
years.all <- c(years, "15-19")
fit <- fitINLA(data = data, geo = geo, Amat = mat,
  year_names = years.all, year_range = c(1985, 2019),
  priors = priors, rw = 2,
  is.yearly=TRUE, m = 5, type.st = 4)
# Projection
out <- projINLA(fit, Amat = mat, is.yearly = TRUE)
plot(out, is.yearly=TRUE, is.subnational=TRUE) + ggplot2::ggtitle("Subnational yearly model")
```

```
## End(Not run)
```

```
fitSpace
```

```
Fit INLA models to perform simple space smoothing.
```

Description

This function calculates the direct estimates by region and fit a simple spatial smoothing model to the direct estimates adjusting for survey design.

Usage

```
fitSpace(data, geo, Amat, family, responseVar, strataVar = "strata",
  weightVar = "weights", regionVar = "region",
  clusterVar = "~v001+v002", hyper = NULL, hyper.besag = c(0.5,
  5e-05), hyper.iid = c(0.5, 5e-05), CI = 0.95, FUN = NULL,
  newformula = NULL, timeVar = NULL, time.model = c("rw1", "rw2")[1],
  hyper.time = NULL, type.st = 0)
```

Arguments

| | |
|-------------|--|
| data | data frame with region and strata information. |
| geo | Geo file |
| Amat | Adjacency matrix for the regions |
| family | Link function specification, currently supports 'binomial' (default with logit link function) or 'gaussian'. |
| responseVar | the response variable |
| strataVar | the strata variable |
| weightVar | the weights variable |
| regionVar | Variable name for region, typically 'v024', for older surveys might be 'v101' |
| clusterVar | Variable name for cluster, typically '~v001 + v002' |
| hyper | the vector of two hyper parameters if specified by user |
| hyper.besag | the vector of two hyper parameters for the structured spatial random effects in Gaussian model, if specified by user |
| hyper.iid | the vector of two hyper parameters for the unstructured spatial random effects in Gaussian model, if specified by user |
| CI | the desired posterior credible interval to calculate |
| FUN | the function to transform the posterior draws. Default to be identify function for normal variable and inverse logit transformation for binomial variables |
| newformula | a string of user-specified random effects model to be used in the INLA call |
| timeVar | The variable indicating time period. If set to NULL then the temporal model and space-time interaction model are ignored. |

| | |
|------------|---|
| time.model | the model for temporal trends and interactions. It can be either "rw1" or "rw2". |
| hyper.time | the vector of two hyper parameters for the structured temporal random effects in Gaussian model, if specified by user |
| type.st | can take values 0 (no interaction), or 1 to 4, corresponding to the type I to IV space-time interaction. |

Details

Normal or binary variables are currently supported. For binary variables, the logit transformation is performed on the direct estimates of probabilities, and a Gaussian additive model is fitted on the logit scale using INLA.

Value

| | |
|--------|------------------------------|
| HT | Direct estimates |
| smooth | Spatially smoothed estimates |
| fit | a fitted INLA object |
| geo | input argument |
| Amat | input argument |
| CI | input argument |
| family | input argument |
| FUN | input argument |

See Also

[countrySummary_mult](#), [fitINLA](#)

Examples

```
## Not run:
data(DemoData2)
data(DemoMap2)
fit <- fitSpace(data=DemoData2, geo=DemoMap2$geo,
Amat=DemoMap2$Amat, family="binomial",
responseVar="tobacco.use", strataVar="strata",
weightVar="weights", regionVar="region",
clusterVar = "~clustid+id",
hyper=NULL, CI = 0.95)

## End(Not run)
```

| | |
|-----------|---|
| getBirths | <i>Function to get Births file from DHS .dta files.</i> |
|-----------|---|

Description

Function to get Births file from DHS .dta files.

Usage

```
getBirths(filepath, surveyyear, variables = c("caseid", "v001", "v002",  
      "v004", "v005", "v021", "v022", "v023", "v024", "v025", "v139", "bidx"),  
      strata = c("v024", "v025"))
```

Arguments

| | |
|------------|---|
| filepath | file path of raw .dta file from DHS |
| surveyyear | year of survey |
| variables | vector of variables to be used in obtaining the person-month files. The variables correspond the the DHS recode manual VI. For early DHS data, the variable names may need to be changed. |
| strata | vector of variable names used for strata. If a single variable is specified, then that variable will be used as strata indicator. If multiple variables are specified, the interaction of these variables will be used as strata indicator. |

Value

A list of birth-month data

Examples

```
## Not run:  
my_fp <- "/myExampleFilepath/surveyData.DTA"  
DemoData <- getBirths(filepath = my_fp, surveyyear = 2015)  
  
## End(Not run)
```

| | |
|------------|---------------------------|
| KingCounty | <i>Map of King County</i> |
|------------|---------------------------|

Description

Shapefiles are King County in the Washington States.

Usage

KingCounty

Format

An object of class SpatialPolygonsDataFrame with 48 rows and 9 columns.

| | |
|-------|-----------------------------|
| logit | <i>Logit transformation</i> |
|-------|-----------------------------|

Description

Logit transformation

Usage

logit(x)

Arguments

x data

Value

logit of x

Examples

```
x <- .5
logit(x)
```

| | |
|---------|------------------------|
| mapPlot | <i>Makes map plot.</i> |
|---------|------------------------|

Description

This function visualizes the map with different variables. The input data frame can be either the long or wide format.

Usage

```
mapPlot(data, variables, values = NULL, labels = NULL, geo, by.data,
        by.geo, is.long = FALSE)
```

Arguments

| | |
|-----------|--|
| data | a data frame with variables to be plotted |
| variables | vector of variables to be plotted. If long format of data is used, only one variable can be selected |
| values | the column corresponding to the values to be plotted, only used when long format of data is used |
| labels | vector of labels to use for each variable, only used when wide format of data is used |
| geo | geo output from read_shape |
| by.data | column name specifying region names in the data |
| by.geo | variable name specifying region names in the data |
| is.long | logical indicator of whether the data is in the long format, default to FALSE |

Examples

```
## Not run:
data(DemoMap)
# Plotting data in the long format
dat <- data.frame(region = rep(c("central", "eastern", "northern", "western"), 3),
  year = rep(c(1980, 1990, 2000), each = 4),
  values = stats::rnorm(12))
utils::head(dat)
mapPlot(dat, variables = "year", values = "values",
  by.data = "region", geo = DemoMap$geo,
  by.geo = "NAME_final", is.long = TRUE)
dat <- data.frame(region = c("central", "eastern", "northern", "western"),
  Year1 = stats::rnorm(4), Year2 = stats::rnorm(4),
  Year3 = stats::rnorm(4))
utils::head(dat)
mapPlot(dat, variables = c("Year1", "Year2", "Year3"),
  labels = c(1980, 1990, 2000),
  by.data = "region", geo = DemoMap$geo,
```

```
by.geo = "NAME_final", is.long = FALSE)
```

```
## End(Not run)
```

| | |
|---------------|--------------------------------|
| plot.projINLA | <i>Plot projection output.</i> |
|---------------|--------------------------------|

Description

Plot projection output.

Usage

```
## S3 method for class 'projINLA'
plot(x, years_label = c("85-89", "90-94", "95-99",
  "00-04", "05-09", "10-14", "15-19"), years_med = c(1987, 1992, 1997,
  2002, 2007, 2012, 2017), is.yearly = TRUE, is.subnational = TRUE,
  proj_year = 2015, ...)
```

Arguments

| | |
|----------------|---|
| x | output from projINLA |
| years_label | labels for the periods |
| years_med | labels for the middle years in each period |
| is.yearly | logical indicator of whether the data contains yearly estimates |
| is.subnational | logical indicator of whether the data contains subnational estimates |
| proj_year | The first year where projections are made, i.e., where no data are available. |
| ... | optional arguments, see details |

Details

Note that arguments after ... must match exactly.

- years_label string of year labels, defaults to c("85-89", "90-94", "95-99", "00-04", "05-09", "10-14", "15-19")
- proj_year projection year as numeric, defaults to 2015
- years_med median of year intervals, defaults to c(1987, 1992, 1997, 2002, 2007, 2012, 2017)
- is.yearly indicator for yearly model, defaults to TRUE
- is.subnational indicator for subnational model, defaults to TRUE

Examples

```

## Not run:
data(DemoData)
deta(DemoMap)
years <- levels(DemoData[[1]]$time)

# obtain direct estimates
data <- countrySummary_mult(births = DemoData,
years = years, idVar = "id",
regionVar = "region", timeVar = "time",
clusterVar = "~clustid+id",
ageVar = "age", weightsVar = "weights",
geo.recode = NULL)

# obtain maps
geo <- DemoMap$geo
mat <- DemoMap$Amat

# Simulate hyper priors
priors <- simhyper(R = 2, nsamp = 1e+05, nsamp.check = 5000, Amat = mat, only.iid = TRUE)

# combine data from multiple surveys
data <- aggregateSurvey(data)

# Model fitting with INLA
years.all <- c(years, "15-19")
fit <- fitINLA(data = data, geo = geo, Amat = mat,
year_names = years.all, year_range = c(1985, 2019),
priors = priors, rw = 2, is.yearly=TRUE,
m = 5, type.st = 4)
# Projection
out <- projINLA(fit, Amat = mat, is.yearly = TRUE)
plot(out, is.yearly=TRUE, is.subnational=TRUE) + ggplot2::ggtitle("Subnational yearly model")

## End(Not run)

```

projINLA

Function to obtain projected estimates from INLA for each time and region.

Description

Function to obtain projected estimates from INLA for each time and region.

Usage

```

projINLA(inla_mod, is.yearly = TRUE, year_range = c(1985, 2019),
year_label = c("85-89", "90-94", "95-99", "00-04", "05-09", "10-14",
"15-19"), Amat = NULL, nsim = 1000)

```

Arguments

| | |
|------------|--|
| inla_mod | output from fitINLA |
| is.yearly | indicator for whether model is yearly or not |
| year_range | range corresponding to year label |
| year_label | vector of year string vector |
| Amat | adjacency matrix |
| nsim | number of simulations |

Value

Results from RW2 model fit, including projection.

Examples

```
## Not run:
years <- levels(DemoData[[1]]$time)

# obtain direct estimates
data <- countrySummary_mult(births = DemoData,
  years = years, idVar = "id",
  regionVar = "region", timeVar = "time",
  clusterVar = "~clustid+id",
  ageVar = "age", weightsVar = "weights",
  geo.recode = NULL)

# obtain maps
geo <- DemoMap$geo
mat <- DemoMap$Amat

# Simulate hyper priors
priors <- simhyper(R = 2, nsamp = 1e+05, nsamp.check = 5000, Amat = mat, only.iid = TRUE)

# combine data from multiple surveys
data <- aggregateSurvey(data)

# Model fitting with INLA
years.all <- c(years, "15-19")
fit <- fitINLA(data = data, geo = geo, Amat = mat,
  year_names = years.all, year_range = c(1985, 2019),
  priors = priors, rw = 2, is.yearly=TRUE,
  m = 5, type.st = 4)
# Projection
out <- projINLA(fit, Amat = mat, is.yearly = TRUE)
plot(out, is.yearly=TRUE, is.subnational=TRUE) + ggplot2::ggtitle("Subnational yearly model")

## End(Not run)
```

| | |
|------------|--------------------------------------|
| read_shape | <i>Function to read shape files.</i> |
|------------|--------------------------------------|

Description

Function to read shape files.

Usage

```
read_shape(filepath, regionnames, data = NULL)
```

Arguments

| | |
|-------------|---|
| filepath | file path for .shp files |
| regionnames | vector of strings of final region names |
| data | optional country summary data, for checking |

Value

A list including shape files and the adjacency matrix.

Examples

```
## Not run:  
my_region_names <- c("central", "eastern", "northern", "western")  
my_fp <- "myExampleFilepath/sdr_subnational_boundaries.shp"  
my_map <- read_shape(filepath = my_fp, regionnames = my_region_names)  
  
## End(Not run)
```

| | |
|-----|---|
| rst | <i>Simulate spatial and temporal random effects</i> |
|-----|---|

Description

This function simulates spatial and temporal random effects with mean zero. The method is described in Algorithm 3.1 of Rue & Held 2015.

Usage

```
rst(n = 1, type = c("s", "t", "st")[1], type.s = "ICAR",  
    type.t = c("RW1", "RW2")[2], Amat = NULL, n.t = NULL,  
    scale.model = TRUE)
```

Arguments

| | |
|-------------|--|
| n | sample size |
| type | type of random effects: temporal (t), spatial (s), or spatial-temporal (st) |
| type.s | type of spatial random effect, currently only ICAR is available |
| type.t | type of temporal random effect, currently only RW1 and RW2 are available |
| Amat | adjacency matrix for the spatial regions |
| n.t | number of time points for the temporal random effect |
| scale.model | logical indicator of whether to scale the random effects to have unit generalized variance. See Sørbye 2013 for more details |

Value

a matrix (for spatial or temporal) or a three-dimensional array (for spatial-temporal) of the random effects.

References

- Rue, H., & Held, L. (2005). *Gaussian Markov random fields: theory and applications*. CRC press.
- Sørbye, S. H. (2013). *Tutorial: Scaling IGMRF-models in R-INLA*. Department of Mathematics and Statistics, University of Tromsø.

Examples

```
## Not run:
data(DemoMap)
## Spatial random effects
out <- rst(n=10000, type = "s", Amat = DemoMap$Amat)
# To verify the mean under the conditional specification
mean(out[,1] - apply(out[,c(2,3,4)], 1, mean))
mean(out[,2] - apply(out[,c(1,3)], 1, mean))
mean(out[,3] - apply(out[,c(1,2,4)], 1, mean))
mean(out[,4] - apply(out[,c(1,3)], 1, mean))

## Temporal random effects (RW1)
out <- rst(n=1, type = "t", type.t = "RW1", n.t = 200, scale.model = FALSE)
par(mfrow = c(1,2))
plot(1:dim(out)[2], out, col = 1, type = "l", xlab = "Time", ylab = "Random effects")
# verify the first order difference is normally distributed
first_diff <- diff(as.numeric(out[1,]))
qqnorm(first_diff )
abline(c(0,1))

## Temporal random effects (RW2)
out <- rst(n=1, type = "t", type.t = "RW2", n.t = 200, scale.model = FALSE)
par(mfrow = c(1,2))
plot(1:dim(out)[2], out, col = 1, type = "l", xlab = "Time", ylab = "Random effects")
# verify the second order difference is normally distributed
first_diff <- diff(as.numeric(out[1,]))
second_diff <- diff(first_diff)
```

```

qqnorm(second_diff)
abline(c(0,1))

## Spacial-temporal random effects
out <- rst(n=1, type = "st", type.t = "RW1", Amat = DemoMap$Amat, n.t = 50)
dimnames(out)
par(mfrow = c(1,1))
plot(1:dim(out)[3], out[1,1,], col = 1,
     type = "l", ylim = range(out), xlab = "Time", ylab = "Random effects")
for(i in 2:4) lines(1:dim(out)[3], out[1,i,], col = i)
legend("bottomright", colnames(DemoMap$Amat), col = c(1:4), lty = rep(1,4))

## End(Not run)

```

simhyper

Function to simulate hyperpriors from an adjacency matrix.

Description

Function to simulate hyperpriors from an adjacency matrix.

Usage

```

simhyper(R = 2, nsamp = 1e+05, nsamp.check = 5000, Amat,
         nperiod = 6, only.iid = TRUE)

```

Arguments

| | |
|-------------|--|
| R | Desired prior odds ratio. Default to 2, i.e., a 95% prior interval for the residual odds ratios lies in the interval (R, 1/R). |
| nsamp | Sample to simulate for scaling factor |
| nsamp.check | Sample to simulate for checking range |
| Amat | Adjacency matrix of the areas in the data. |
| nperiod | numerical value of how many time periods in the data |
| only.iid | Indicator for whether or not only IID hyperpriors are simulated |

References

Wakefield, J. Multi-level modelling, the ecologic fallacy, and hybrid study designs. *International Journal of Epidemiology*, 2009, vol. 38 (pg. 330-336).

Examples

```
## Not run:  
data(DemoMap)  
mat <- DemoMap$Amat  
priors <- simhyper(R = 2, nsamp = 1e+05, nsamp.check = 5000, Amat = mat)  
  
## End(Not run)
```

Index

*Topic **datasets**

- BRFSS, [4](#)
- DemoData, [7](#)
- DemoData2, [8](#)
- DemoMap, [8](#)
- DemoMap2, [9](#)
- KingCounty, [15](#)

aggregateSurvey, [3](#)

BRFSS, [4](#)

ChangeRegion, [4](#), [5](#), [6](#)

countrySummary, [5](#), [7](#), [11](#)

countrySummary_mult, [3](#), [6](#), [6](#), [13](#)

DemoData, [7](#)

DemoData2, [8](#)

DemoMap, [8](#)

DemoMap2, [9](#)

expit, [9](#)

fitINLA, [10](#), [13](#), [19](#)

fitSpace, [12](#)

getBirths, [5](#), [6](#), [14](#)

KingCounty, [15](#)

logit, [15](#)

mapPlot, [16](#)

plot.projINLA, [17](#)

projINLA, [17](#), [18](#)

read_shape, [16](#), [20](#)

rst, [20](#)

simhyper, [10](#), [22](#)

SUMMER (SUMMER-package), [2](#)

SUMMER-package, [2](#)