

Package ‘SUMMER’

April 20, 2018

Type Package

Title Spatio-Temporal Under-Five Mortality Methods for Estimation

Version 0.2.0

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 RColorBrewer, classInt, rgeos, maptools, survey, grDevices, graphics, stats, spdep, survival, ggplot2, sp, utils, Matrix, reshape2, pixmap

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

Additional_repositories <https://www.math.ntnu.no/inla/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 2018-04-20 21:56:18 UTC

R topics documented:

SUMMER-package	2
aggregateSurvey	3
ChangeRegion	4
countrySummary	5
countrySummary_mult	6
DemoData	7
DemoData2	7
DemoMap	8
DemoMap2	8
expit	9
fitINLA	9
fitSpace	11
getBirths	13
logit	14
mapPlot	14
plot.projINLA	15
projINLA	17
read_shape	18
simhyper	19
Index	20

SUMMER-package	<i>SUMMER package documentation.</i>
----------------	--------------------------------------

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)
```

ChangeRegion	<i>Function to map region names to a common set.</i>
--------------	--

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

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

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	<i>Fake dataset for vignette.</i>
----------	-----------------------------------

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	<i>Fake dataset for vignette.</i>
-----------	-----------------------------------

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

<code>data</code>	Combined dataset
<code>Amat</code>	Adjacency matrix for the regions
<code>geo</code>	Geo file
<code>formula</code>	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.
<code>rw</code>	Take values 1 or 2, indicating the order of random walk.
<code>is.yearly</code>	Logical indicator for fitting yearly or period model.
<code>year_names</code>	string vector of year names
<code>year_range</code>	Entire range of the years (inclusive) defined in <code>year_names</code> .
<code>m</code>	Number of years in each period.
<code>na.rm</code>	Logical indicator of whether to remove rows with NA values in the data. Default set to TRUE.
<code>redo.prior</code>	Logical indicator of whether to re-estimate hyperparameters
<code>priors</code>	priors from simhyper
<code>type.st</code>	type for space-time interaction
<code>useHyper</code>	option to manually set all hyperpriors
<code>a.iid</code>	hyperparameter for i.i.d random effects, only need if <code>useHyper = TRUE</code>
<code>b.iid</code>	hyperparameter for i.i.d random effects, only need if <code>useHyper = TRUE</code>
<code>a.rw1</code>	hyperparameter for RW1 random effects, only need if <code>useHyper = TRUE</code>
<code>b.rw1</code>	hyperparameter for RW1 random effects, only need if <code>useHyper = TRUE</code>
<code>a.rw2</code>	hyperparameter for RW2 random effects, only need if <code>useHyper = TRUE</code>
<code>b.rw2</code>	hyperparameter for RW2 random effects, only need if <code>useHyper = TRUE</code>
<code>a.icar</code>	hyperparameter for ICAR random effects, only need if <code>useHyper = TRUE</code>
<code>b.icar</code>	hyperparameter for ICAR random effects, only need if <code>useHyper = TRUE</code>

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, CI = 0.95, FUN = NULL)

```

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
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

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*Function to get Births file from DHS .dta files.*

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)
```

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

Plot projection output.

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	<i>Function to obtain projected estimates from INLA for each time and region.</i>
----------	---

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

Function to read shape files.

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)
```

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**

DemoData, [7](#)

DemoData2, [7](#)

DemoMap, [8](#)

DemoMap2, [8](#)

aggregateSurvey, [3](#)

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

countrySummary, [5](#), [6](#), [10](#)

countrySummary_mult, [3](#), [5](#), [6](#), [12](#)

DemoData, [7](#)

DemoData2, [7](#)

DemoMap, [8](#)

DemoMap2, [8](#)

expit, [9](#)

fitINLA, [9](#), [12](#), [17](#)

fitSpace, [11](#)

getBirths, [5](#), [6](#), [13](#)

logit, [14](#)

mapPlot, [14](#)

plot.projINLA, [15](#)

projINLA, [16](#), [17](#)

read_shape, [15](#), [18](#)

simhyper, [10](#), [19](#)

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

SUMMER-package, [2](#)