

Package ‘spacom’

July 2, 2014

Type Package

Title Spatially weighted context data for multilevel modelling

Version 1.0-4

Date 2013-11-26

Description The package provides tools to construct and exploit spatially weighted context data. Spatial weights are derived by a Kernel function from a user-defined matrix of distances between contextual units. Spatial weights can then be applied either to precise contextual measures or to aggregate estimates based on micro-level survey data, to compute spatially weighted context data. Available aggregation functions include indicators of central tendency, dispersion, or inter-group variability, and take into account survey design weights. The package further allows combining the resulting spatially weighted context data with individual-level predictor and outcome variables, for the purposes of multilevel modelling. An ad hoc stratified bootstrap resampling procedure generates robust point estimates for multilevel regression coefficients and model fit indicators, and computes confidence intervals adjusted for measurement dependency and measurement error of aggregate estimates. As an additional feature, residual and explained spatial dependency can be estimated for the tested models.

License GPL (>= 2)

LazyLoad yes

Depends R (>= 2.14-0)

Imports methods, spdep, foreach, iterators, lme4, nlme, Matrix

Collate all_classes.R common.R moran.R SpawML.R Descriptives.R
ResExSpawML.R ResAggSpawML.R Weights.R ExplorSpawML.R

Author Till Junge [aut, cre], Sandra Penic [aut], Mathieu Cos-
suta [aut], Guy Elcheroth [aut], Stephanie Glaeser [ctb], Davide Morselli [ctb]

Maintainer Till Junge <till.junge@altermail.ch>

NeedsCompilation no

Repository CRAN

Date/Publication 2013-12-17 16:46:02

R topics documented:

spacom-package	2
aggregation functions	5
CH1903	8
d_ethnic	8
d_geo	9
d_ident	9
d_migr	10
ExploreMLSpawExact	10
GPS	13
homog_census	14
MLSpawExact	14
MLSpawExactOutput-class	16
MLSpawResidMoran	18
ResampleMLSpawAggregate	20
ResampleMLSpawExact	23
ResampleMLSpawOutput-class	25
SpawAggregate	27
SpawAggregateOutput-class	30
SpawExact	32
traces_event	33
traces_ind	34
WeightMatrix	35
Index	37

spacom-package

Spatially weighted context data for multilevel modelling

Description

The package provides tools to construct and exploit spatially weighted context data. Spatial weights are derived by a Kernel function from a user-defined matrix of distances between contextual units. Spatial weights can then be applied either to precise contextual measures or to aggregate estimates based on micro-level survey data, to compute spatially weighted context data. Available aggregation functions include indicators of central tendency, dispersion, or inter-group variability, and take into account survey design weights. The package further allows combining the resulting spatially weighted context data with individual-level predictor and outcome variables, for the purposes of multilevel modelling. An ad hoc stratified bootstrap resampling procedure generates robust point estimates for multilevel regression coefficients and model fit indicators, and computes confidence intervals adjusted for measurement dependency and measurement error of aggregate estimates. As an additional feature, residual and explained spatial dependency can be estimated for the tested models.

Details

Package: spacom
Type: Package
Version: 1.0-0
Date: 2012-12-21
Authors: Till Junge
Sandra Penic
Mathieu Cossuta
Guy Elcheroth
Contributors: Davide Morselli
Stephanie Glaeser
Maintainer: Till Junge <till.junge@gmail.com>
License: GPL (>= 2)
LazyLoad: yes
Depends: methods, R (>= 2.14-0), spdep, foreach, lme4, boot, classInt, iterators, rgeos

The package is structured by seven core functions, organised along four logical steps. In each of the first three steps three types of input data required for multilevel modelling with spatially weighted context data are successively imported and transformed, to generate output objects required in subsequent steps:

Step 1: Creation of spatial weighting matrices. User-specified Kernel functions and bandwidth values (function [WeightMatrix](#)) are applied to one or several user-defined distance matrices.

Step 2: Construction of spatially weighted context data. The spatial weights generated in step 1 are combined with (a) imported precise contextual measures (function [SpawExact](#)) and/or (b) imported micro-level survey data to compute contextual estimates through a user-specified aggregation function (function [SpawAggregate](#)).

Step 3: Multilevel modelling with spatially weighted context data. Imported individual-level predictor and outcome variables are combined with the precise (functions [MLSpawExact](#) and [ResampleMLSpawExact](#)) or aggregate (function [ResampleMLSpawAggregate](#)) context data generated in step 2.

Finally, contextual residuals from multilevel models can be re-used in an additional, fourth step to compute estimates of explained spatial dependency (function [MLSpawResidMoran](#)).

The package furthermore provides a fast-track procedure, which condenses steps 1 to 3 and directly generates point estimates for regression coefficients across a range of different bandwidth values, in order to explore the scale-dependency of contextual effects prior to more intensive analyses (function [ExploreMLSpawExact](#))

Author(s)

Till Junge, Sandra Penic, Mathieu Cossuta, Guy Elcheroth

Maintainer: Till Junge <till.junge@gmail.com> Contributors: Stephanie Glaeser, Davide Morselli

References

Elcheroth, G., Penic, S., Fasel, R., Giudici, F., Glaeser, S., Joye, D., Le Goff, J.-M., Morselli, D., & Spini, D. (2012). Spatially weighted context data: a new approach for modelling the impact of

collective experiences. *LIVES Working Papers*, 19.

Examples

```
## Testing the model described in Elcheroth et al (2012), page 25, Equation 9.4
## Collective guilt assignment (outcome) is predicted by a series of individual level predictors,
## and two contextual predictors: risk of war victimization weighted by geographical proximity and
## common territorial identification

## DATA PREPARATION
## Import data
## load individual level data
data(traces_ind)

## exclude variable "cg_acc" and missing values
traces_ind <- na.exclude(traces_ind[,-6])

## load contextual indicator for aggregation
data(traces_event)

## load distance matrices
data(d_geo)
data(d_ident)

## EXPLORATORY ANALYSIS

## Examining the impact of the risk of war victimization weighted by geographical
## proximity weights with bandwidths of 25, 50 and 100 km

explo.wv <- ExploreMLSpawAggregate(individual.level.data=traces_ind,
                                   contextual.name="w_all",
                                   contextual.data=traces_event,
                                   context.id="area.name",
                                   formula=cg_ass~victim_d+comb_d+
                                   male+age_1990+high_school+higher_edu+
                                   (1|area.name),
                                   distance.matrix=d_geo,
                                   multilevel.bandwidths=c(25, 50, 100),
                                   design.weight.names = "weight",
                                   aggregation.function = "weighted.mean",
                                   kernel = NULL)

## STEP 1: Create spatial weights

## geographical proximity, bandwidth=50
geow.50 <- WeightMatrix(d_geo, bandwidth=50)

## common territorial identification, bandwidth=2
w.id <- WeightMatrix(d_ident,bandwidth=2)

## STEP 2: Create spatially weighted aggregated indicator of risk of war victimization
```

```

wv.agg <- SpawAggregate(contextual.data=traces_event,
                       context.id="area.name",
                       contextual.names=c("w_all", "w_all"),
                       contextual.weight.matrices=list(geow.50, w.id),
                       aggregation.functions="weighted.mean",
                       design.weight.names="weight",
                       nb.resamples=2)
## If wanted, the names of aggregated indicators in wv.agg can be changed
names(wv.agg) <- c("wv.50", "wv.id")

## STEP 3: Perform multilevel analysis (Eq 9.4, Elcheroth et al (2012), p. 25)

m1a.results <- ResampleMLSpawAggregate(individual.level.data=traces_ind,
                                       context.id="area.name",
                                       formula= cg_ass ~ victim_d+ comb_d+ male+ age_1990+
                                               high_school+ higher_edu + (1|area.name)+ wv.50 + wv.id,
                                       aggregates = wv.agg, precise.data=NULL)

## STEP 4: Test the residual spatial dependency of the previous model
m1a.results.moran <- MLSpawResidMoran(ml.spaw.obj=m1a.results,
                                       distance.matrix=d_geo,
                                       bandwidths=c(25,50,100,200))

## Plotting spatial variogram with the values in m1a.results.moran
## See plot() and par() for details
## Not run:
plot(m1a.results.moran[,1], xaxt="n", xlab="Bandwidth values", ylab="Moran's I", type="b")
axis(side=1, at=1:nrow(m1a.results.moran), labels=rownames(m1a.results.moran))
## End(Not run)

```

aggregation functions *Aggregation functions provided by spacom*

Description

Spacom provides a set of aggregation functions which can be used with the functions [SpawAggregate](#) and [ExploreMLSpawAggregate](#). See **Details** for descriptions of each function. Note that you are not restricted to these.

User supplied aggregation functions can be used if they have the form

$$x_w = f(x, w, \dots)$$

where x is the data to be aggregated, w the weights to be applied. Any number of additional arguments may also be used (see argument `additional.args` in [SpawAggregate](#) for details on additional arguments. The return value x_w must be a scalar value (i.e. not a vector).

Usage

```

wt.sd(data, weights=rep(1, length(data)))
wt.var(data, weights=rep(1, length(data)))
wt.gini(data, weights=rep(1, length(data)))
wt.gini.categ(data, weights=rep(1, length(data)))
wt.Theil(data, weights=rep(1, length(data)))
wt.RS(data, weights=rep(1, length(data)))
wt.Atkinson(data, weights=rep(1, length(data)))
wt.gini.group(data, weights=rep(1, length(data)), groups)

```

Arguments

`data` A vector of data to be aggregated

`weights` A vector of weights. These weights are computed by combining the spatial weights and design weights for each context.

`groups` name of the column used to group data (only `wt.group.gini`)

Details

1. `wt.var(data, weights)` Computes the weighted variance according to

$$\text{Var}(x) = \frac{\sum_{i=1}^n (w_i \cdot (x_i - \mu)^2)}{\sum_{i=1}^n w_i}$$

where the weighted mean is μ

$$\mu = \frac{\sum_{i=1}^n w_i \cdot x_i}{\sum_{i=1}^n w_i}$$

2. `wt.sd(data, weights)` Computes the weighted standard deviation according to

$$\sigma = \sqrt{\text{Var}(x)}$$

3. `wt.gini` The weighted Gini coefficient is given by

$$G = \frac{1}{2\mu} \sum_{i,j} w_i w_j |x_i - x_j|$$

where μ is

$$\mu = \sum_i w_i x_i$$

4. `wt.gini.categ` Spacom also allows for the computation of inequality indicators for categorical variables. In this case, one has no x_i value but only the weighted frequencies f_i for each category. For that case, the Gini becomes

$$G = 1 - \sum_i f_i^2$$

The weighted frequencies are computed from data and weights

5. `wt.gini.group` get from guy

Value

A scalar value of type numeric. For user-supplied aggregation functions, this can potentially be of a different type.

Author(s)

Mathieu Cossuta, Till Junge, Sandra Penic, Guy Elcheroth

References

Elcheroth, G., Penic, S., Fasel, R., Giudici, F., Glaeser, S., Joye, D., Le Goff, J.-M., Morselli, D., & Spini, D. (2012). Spatially weighted context data: a new approach for modelling the impact of collective experiences. *LIVES Working Papers*, 19.

Examples

```
## Creating indicators of spatially weighted inequalities

## load individual level data and remove missings
data(traces_ind)
traces_ind <- na.exclude(traces_ind)

## create variable "simulated income" in the dataset
## the variable is created only in demonstrative purposes,
## and it is not part of the TRACES dataset

traces_ind$sim_inc <- rnorm(nrow(traces_ind), 1000, 100)

## Step 1: Load distance matrix and create weights
data(d_geo)
geow.50 <- WeightMatrix(d_geo, bandwidth=50)

## Step 2: Spatially weighted gini for simulated income
si.gini <- SpawAggregate(contextual.data=traces_ind,
                        context.id="area.name",
                        contextual.names="sim_inc",
                        contextual.weight.matrices=geow.50,
                        aggregation.functions="wt.gini",
                        design.weight.names=NULL,
                        nb.resamples=5)

## Step 2: Spatially weighted gini for groups (Spatially weighted inequalities
## in simulated income for men and women)

si.gini.gr <- SpawAggregate(contextual.data=traces_ind,
                           context.id="area.name",
                           contextual.names="sim_inc",
                           contextual.weight.matrices=geow.50,
                           aggregation.functions="wt.gini.group",
                           additional.args="male",
                           design.weight.names=NULL,
                           nb.resamples=5)
```

 CH1903

 CH1903

Description

Swiss square projection

Usage

```
data(CH1903)
```

Format

The format is: Formal class 'CRS' [package "sp"] with 1 slots ..@ projargs: chr " +init=world:CH1903 +proj=somerc +lat_0=46d57'8.660"N +lon_0=7d26'22.500"E +ellps=bessel +x_0=600000 +y_0=200000 +k_0=1. +no_"|__truncated__

 d_ethnic

A distance matrix for TRACES areas

Description

A matrix of dimension 80×80 , containing distances d_{ij} between centers of TRACES areas i and j defined as dissimilarity in the ethnic composition of a population.

Usage

```
d_ethnic
```

Source

Elcheroth, G., Penic, S., Fasel, R., Giudici, F., Glaeser, S., Joye, D., Le Goff, J.-M., Morselli, D., & Spini, D. (2012). Spatially weighted context data: a new approach for modelling the impact of collective experiences. *LIVES Working Papers*, 19.

Examples

```
data(d_ethnic)
w.eth <- WeightMatrix(distance.matrix=d_ethnic, bandwidth=1)
```

d_geo	<i>A distance matrix for TRACES areas</i>
-------	---

Description

A matrix of dimension 80×80 , containing geographical distances (in km) d_{ij} between centers of TRACES areas i and j .

Usage

```
d_geo
```

Source

Elcheroth, G., Penic, S., Fasel, R., Giudici, F., Glaeser, S., Joye, D., Le Goff, J.-M., Morselli, D., & Spini, D. (2012). Spatially weighted context data: a new approach for modelling the impact of collective experiences. *LIVES Working Papers, 19*.

Examples

```
data(d_geo)
geow.50 <- WeightMatrix(distance.matrix=d_geo, bandwidth=50)
```

d_ident	<i>A distance matrix for TRACES areas</i>
---------	---

Description

A matrix of dimension 80×80 , containing distances d_{ij} between centers of TRACES areas i and j defined as a lack of common identification.

Usage

```
d_ident
```

Source

Elcheroth, G., Penic, S., Fasel, R., Giudici, F., Glaeser, S., Joye, D., Le Goff, J.-M., Morselli, D., & Spini, D. (2012). Spatially weighted context data: a new approach for modelling the impact of collective experiences. *LIVES Working Papers, 19*.

Examples

```
data(d_ident)
w.id <- WeightMatrix(distance.matrix=d_ident, bandwidth=2)
```

d_migr	<i>A distance matrix for TRACES areas</i>
--------	---

Description

A matrix of dimension 80×80 , containing distances d_{ij} between centers of TRACES areas i and j defined as a lack of contact between two populations.

Usage

```
d_migr
```

Source

Elcheroth, G., Penic, S., Fasel, R., Giudici, F., Glaeser, S., Joye, D., Le Goff, J.-M., Morselli, D., & Spini, D. (2012). Spatially weighted context data: a new approach for modelling the impact of collective experiences. *LIVES Working Papers*, 19.

Examples

```
data(d_migr)
w.migr <- WeightMatrix(distance.matrix=d_migr, bandwidth=25)
```

ExploreMLSpawExact	<i>Exploratory multilevel analysis with spatially weighted context data</i>
--------------------	---

Description

These exploratory functions provide fast-track procedures, which condense steps 1 to 3. On the basis of precise contextual measures and/or micro-level survey data for estimating contextual indicators, a distance matrix between contextual units, and individual outcomes and (optional) predictor variables, it directly generates point estimates for regression coefficients of a user-defined multilevel model, for a range of different bandwidth values. This function can be used to first explore the scale of relevant contextual effects, before parametrising the model more precisely, calculating (computationally expensive) confidence intervals and estimating spatial dependency for specific models, on a step-by-step basis.

Usage

```
ExploreMLSpawExact(individual.level.data,
  contextual.name,
  context.id,
  formula,
  distance.matrix,
  multilevel.bandwidths,
  precise.data,
```

```

kernel = NULL,
verbose = TRUE)

ExploreMLSpawAggregate(individual.level.data,
contextual.name,
contextual.data,
context.id, formula,
distance.matrix,
multilevel.bandwidths,
design.weight.names = NULL,
aggregation.function = "weighted.mean",
kernel = NULL,
additional.args = NULL,
verbose = TRUE)

```

Arguments

`individual.level.data`
A data.frame containing the individual level predictor and outcome variables. The data must be numeric (logical, integer, or plain numeric). There must not be any missing values, NaNs, NULLs or NAs.

`contextual.name`
A name of contextual variable to be weighted.

`contextual.data`
A data.frame consisting of micro-level data to generate contextual indicators by aggregation and containing a column named `context.id` with the context ID variable. This is an individual level dataframe. The data must be numeric (logical, integer, or plain numeric). There must not be any missing values, NaNs, NULLs or NAs. By default set to NULL.

`context.id`
The name of the context ID variable. This variable allows matching contextual units from different data sets (`contextual.data`, `individual.level.data`, `distance.matrix`). Values can be of any type but need to be defined consistently across data sets.

`formula`
Formula description of the model. The formula is handed down to `lmer` function of the package `lme4`. Refer to the documentation of `lmer` for details.

`distance.matrix`
A square matrix of dimension $n \times n$, where n is the number of contextual units.

`multilevel.bandwidths`
A vector of scalar numeric values specifying the bandwidth h

`precise.data`
A data.frame containing the precise contextual data. There must not be any missing values, NaNs, NULLs or NAs. By default set to NULL.

`design.weight.names`
A name of optional design weight at the individual level used for aggregation (for example, for a weighted mean). By default set to NULL.

`aggregation.function`
A name of aggregation function. Function takes either

	<ol style="list-style-type: none"> 1 argument in which case the corresponding individual design weight is NULL, 2 arguments in which case the second argument is taken from the corresponding individual design weight. Defaults to "mean".
kernel	<p>A function applied to the distance matrix. By default NULL, in which case the kernel function</p> $w_{ij} = f(d, h) = \left(\frac{1}{2}\right)^{d_{ij}^2/h^2}$ <p>is used, where w_{ij}, d_{ij}, h are elements of the weight matrix \mathbf{W}, of the distance matrix \mathbf{D} and the bandwidth h. User-supplied kernel functions have to take 2 arguments and return a matrix of the same dimension as the first argument.</p>
additional.args	For aggregation functions which take additional arguments (that is in addition to the data to aggregate and design weights), they can be specified here. <code>additional.args</code> defaults to NULL
verbose	if TRUE, progress information and an estimate of the remaining computing time is shown at the screen. Defaults to TRUE

Details

ExploreMLSpawExact performs exploratory multilevel analysis with a precise spatially weighted contextual indicator. ExploreMLSpawAggregate performs analysis with an aggregated spatially weighted contextual indicator. Both functions provide only conventional statistical estimates and accept only one contextual predictor.

Value

A list of `MLSpawExactOutput-class` objects

Author(s)

Till Junge, Sandra Penic, Guy Elcheroth

References

Elcheroth, G., Penic, S., Fasel, R., Giudici, F., Glaeser, S., Joye, D., Le Goff, J.-M., Morselli, D., & Spini, D. (2012). Spatially weighted context data: a new approach for modelling the impact of collective experiences. *LIVES Working Papers*, 19.

Examples

```
#### ExploreMLSpaw
# Exploratory spatially weighted multilevel with standard estimates of
# standard errors. Accepts only one contextual predictor. Predicting
# collective guilt acceptance. Precise contextual predictor (ethnic homogeneity)
# weighted with geographical proximity weights, h=50,100,200.

# load individual level data, remove collective guilt assignment from the
# data frame, and remove NA's
```

```
data(traces_ind)

traces_ind <- traces_ind[,-7]
traces_ind <- na.exclude(traces_ind)

# load precise contextual indicator
data(homog_census)

# load distance matrix
data(d_geo)

# perform ExploreMLSpawExact
acc_homog <- ExploreMLSpawExact(
  individual.level.data=traces_ind,
  contextual.name="Homog_00",
  context.id="area.name",
  formula=cg_acc ~ victim_d + comb_d + male + age_1990 + high_school +
  higher_edu + (1|area.name),
  precise.data=homog_census,
  distance.matrix=d_geo,
  multilevel.bandwidths=c(50, 100))
```

GPS

GPS

Description

describe me

Usage

```
data(GPS)
```

Format

The format is: Formal class 'CRS' [package "sp"] with 1 slots ..@ projargs: chr "+proj=longlat +datum=WGS84 +ellps=WGS84 +towgs84=0,0,0"

Examples

```
data(GPS)
## maybe str(GPS) ; plot(GPS) ...
```

homog_census	<i>Dataset homog_census</i>
--------------	-----------------------------

Description

A dataset with an indicator of ethnic homogeneity in TRACES areas of former Yugoslavia, calculated on the basis of national censuses and official estimates from 2000 to 2006. It should be used as a precise contextual indicator in multilevel analyses.

Usage

```
homog_census
```

Format

The dataframe contains the following columns:

area - context ID variable, ranges from 1 to 80

Homog_00 - proportion of largest ethnic group in the area

Source

Our own calculation based on national censuses and official estimates from 2000 to 2006.

Examples

```
data(homog_census)
```

MLSpawExact	<i>Standard multilevel analysis applied to spatially weighted context data</i>
-------------	--

Description

Integrates spatially weighted context data as higher-level predictors in multilevel analysis and produces conventional statistical estimates.

Usage

```
MLSpawExact(individual.level.data,
             context.id,
             formula,
             precise.data=NULL,
             verbose=TRUE,
             ...)
```

Arguments

<code>individual.level.data</code>	A data.frame containing the individual level predictor and outcome variables. The data must be numeric (logical, integer, or plain numeric). There must not be any missing values, NaNs, NULLs or NAs.
<code>context.id</code>	The name of the context ID variable. This variable allows matching contextual units from different data sets (<code>precise.data</code> , <code>individual.level.data</code>). Values can be of any type but need to be defined consistently across data sets.
<code>formula</code>	Formula description of the model. The formula is handed down to <code>lmer</code> function of the package <code>lme4</code> . Refer to the documentation of <code>lmer</code> for details.
<code>precise.data</code>	A data.frame containing the precise contextual data. There must not be any missing values, NaNs, NULLs or NAs. By default set to NULL.
<code>verbose</code>	if TRUE, progress information and an estimate of the remaining computing time is shown at the screen. Defaults to TRUE
<code>...</code>	All additional named arguments are handed through to the function <code>lme:lmer</code> (for example, <code>REML=TRUE</code>).

Value

An object of class `MLSpawExactOutput-class`

Author(s)

Till Junge, Sandra Penic, Mathieu Cossuta, Guy Elcheroth

References

Elcheroth, G., Penic, S., Fasel, R., Giudici, F., Glaeser, S., Joye, D., Le Goff, J.-M., Morselli, D., & Spini, D. (2012). Spatially weighted context data: a new approach for modelling the impact of collective experiences. *LIVES Working Papers*, 19.

Examples

```
## Spatially weighted multilevel analysis, with standard estimates of
## standard errors.
## It is step-3 function
## Model with two contextual predictors, treated as precise,
## predicting collective guilt acceptance.

## Data preparation
## load individual level data, remove collective guilt assignment from the
## data frame, and remove NA's
data(traces_ind)

traces_ind <- traces_ind[,-7]
traces_ind <- na.exclude(traces_ind)

## load contextual indicator for aggregation
data(traces_event)
```

```

## load precise contextual indicator
data(homog_census)

## load distance matrix
data(d_geo)

## Step 1: Create spatial weights
geow.100 <- WeightMatrix(d_geo, bandwidth=100)

## Step 2: Create spatially weighted contextual indicators

## from micro-level survey data (treated as precise)
aggregate <- SpawAggregate(contextual.data=traces_event,
                           context.id="area.name",
                           contextual.names='w_all',
                           contextual.weight.matrices=geow.100,
                           aggregation.functions="weighted.mean",
                           design.weight.names="weight",
                           nb.resamples=0)

## from precise indicator
weighted.homog.census <- SpawExact(precise.data=homog_census,
                                   context.id="area.name",
                                   contextual.names="Homog_00",
                                   contextual.weight.matrices=geow.100)

## merge aggregated and weighted data
context.data <- merge(aggregate, weighted.homog.census, by="area.name")

## Step 3: Perform MLSpawExact with two spatially weighted indicators
acc_w_homog_100 <-
  MLSpawExact(individual.level.data=traces_ind,
              context.id="area.name",
              formula=cg_acc ~ victim_d + comb_d + male + age_1990 +
                high_school + higher_edu + (1|area.name) + w_all.1 + Homog_00.1,
              precise.data=context.data)

```

MLSpawExactOutput-class

Class "MLSpawExactOutput"

Description

Output object of the function MLSpawExact.

Objects from the Class

Objects can be created by calls of the form `new("MLSpawExactOutput", ...)`.

Slots

lme: Object of class "merMod"
 beta: Object of class "numeric"

Methods

AIC signature(object = "MLSpawExactOutput"): merMod of lme4
BIC signature(object = "MLSpawExactOutput"): merMod of lme4
fixef signature(object = "MLSpawExactOutput"): merMod of lme4
print signature(x = "MLSpawExactOutput")
ranef signature(object = "MLSpawExactOutput"): merMod of lme4
show signature(object = "MLSpawExactOutput")
summary signature(object = "MLSpawExactOutput")
VarCorr signature(x = "MLSpawExactOutput"): merMod of lme4

Author(s)

Till Junge

Examples

```
## Perform MLSpawExact

## Import and prepare data (see details in MLSpawExact)

data(traces_ind)
traces_ind <- traces_ind[,-7]
traces_ind <- na.exclude(traces_ind)
data(traces_event)

data(homog_census)
data(d_geo)

## Step 1: Create spatial weights
geow.100 <- WeightMatrix(d_geo, bandwidth=100)

## Step 2: Create spatially weighted contextual indicators
aggregate <- SpawAggregate(contextual.data=traces_event,
                           context.id="area.name",
                           contextual.names='w_all',
                           contextual.weight.matrices=geow.100,
                           aggregation.functions="weighted.mean",
                           design.weight.names="weight",
                           nb.resamples=0)

## Step 3: Perform MLSpawExact with the spatially weighted indicator
acc_w100 <- MLSpawExact(individual.level.data=traces_ind,
                        context.id="area.name",
```

```

        formula=cg_acc ~ victim_d + comb_d + male + age_1990 +
        high_school + higher_edu + (1|area.name) + w_all.1,
        precise.data=aggregate)

## acc_w100 is an object of class MLSpawExactOutput
class(acc_w100)

## to assess standardized coefficients
acc_w100@beta

## look at the fitted mixed-effect model :
acc_w100@lme
## to assess non-standardized fixed coefficients
fixef(acc_w100@lme)

## to assess the deviance of the fitted model
deviance(acc_w100@lme)

## for details about slots within @lme see ?merMod-class{lme4}

```

MLSpawResidMoran

Testing the (residual) spatial dependency

Description

This step-4 function uses the contextual residuals of (spatially weighted) multilevel models to estimate the spatial dependency left unexplained by the model. The spatial dependency is operationalised by Moran's I coefficient for spatial auto-correlation, which can be recalculated for a range of different bandwidth values (which, in a similar way to spatial weights used for the creation of the context data, allow to parametrise the scale at which spatial dependency is being estimated). The part of explained spatial dependency can be obtained by comparison with the spatial dependency of the intercept-only, or with any other reference model, at the same bandwidth value.

Usage

```

MLSpawResidMoran(ml.spaw.obj,
                 distance.matrix,
                 bandwidths,
                 kernel = NULL,
                 confidence.intervals = c(0.95),
                 verbose = TRUE)

```

Arguments

<code>ml.spaw.obj</code>	ResampleMLSpawOutput, MLSpawExactOutput, or matrix of residuals
<code>distance.matrix</code>	square matrix of dimension $n \times n$, where n is the number of contextual units.
<code>bandwidths</code>	vector of bandwidths for which to compute Moran's I.

kernel function applied to the distance matrix. By default NULL, in which case the kernel function

$$w_{ij} = f(d, h) = \left(\frac{1}{2}\right)^{d_{ij}^2/h^2}$$

is used, where w_{ij} , d_{ij} , h are elements of the weight matrix \mathbf{W} , of the distance matrix \mathbf{D} and the bandwidth h . User-supplied kernel functions have to take 2 arguments and return a matrix of the same dimension as the first argument.

confidence.intervals

vector of confidence intervals. Defaults to `c(.95)` which corresponds to 95 %.

verbose

if TRUE, progress information and an estimate of the remaining computing time is shown at the screen. Defaults to TRUE

Value

A matrix containing Moran's I's

Author(s)

Till Junge, Sandra Penic, Guy Elcheroth

References

Elcheroth, G., Penic, S., Fasel, R., Giudici, F., Glaeser, S., Joye, D., Le Goff, J.-M., Morselli, D., & Spini, D. (2012). Spatially weighted context data: a new approach for modelling the impact of collective experiences. *LIVES Working Papers*, 19.

Examples

```
# Residual spatial autocorrelation
## It is step-4 function
## for residuals from ResampleAggregateSpawML

## Data preparation

## load individual level data, remove collective guilt assignment for the
## data frame, and remove NA's
data(traces_ind)
traces_ind <- traces_ind[,-7]
traces_ind <- na.exclude(traces_ind)

## load contextual indicator for aggregation
data(traces_event)

## load precise contextual indicator
data(homog_census)

## load distance matrix
data(d_geo)
```

```

## Step 1: Create spatial weights
geow.100 <- WeightMatrix(d_geo, bandwidth=100)

## Step 2: Compute spatially weighted aggregated contextual indicator
wv.agg.100 <- SpawAggregate(
  contextual.data = traces_event,
  context.id="area.name",
  contextual.names = "w_all",
  contextual.weight.matrices=geow.100,
  nb.resamples=5,
  aggregation.functions="weighted.mean",
  design.weight.names="weight",
  sample.seed=1)

## rename weighted variable name so it reflects the used weighting
## matrix
names(wv.agg.100) <- "w_all.100"

## Step 3: Perform ResampleMLSpawAggregate
acc_w_100 <-
ResampleMLSpawAggregate(
  individual.level.data=traces_ind,
  context.id="area.name",
  formula=cg_acc ~ victim_d + comb_d + male + age_1990 + high_school +
  higher_edu + (1|area.name) + w_all.100,
  aggregates=wv.agg.100,
  precise.data=NULL)

## Step 4: Perform MLSpawResidMoran for bandwidths 25, 50, 100, 200

## Not run:
MI_acc <- MLSpawResidMoran(ml.spaw.obj=acc_w_100,
  distance.matrix=d_geo,
  bandwidths=c(25,50,100,200))
## End(Not run)

## The results can be used for plotting spatial variogram
## See plot() and par() for details
## Not run:
plot(MI_acc[,1], xaxt="n", xlab="Bandwidth values", ylab="Moran's I", type="b")
axis(side=1, at=1:nrow(MI_acc), labels=rownames(MI_acc))
## End(Not run)

```

ResampleMLSpawAggregate

Spatially weighted multilevel analysis with stratified resampling for aggregated context data

Description

This step-3 function performs multilevel analyses with spatially weighted context data based on aggregate survey estimates. The context data created in step 2 are combined with additional individual outcome and (optional) predictor variables, to test a user-defined model. An ad hoc stratified bootstrap resampling procedure generates robust point estimates for regression coefficients and model fit indicators, and computes confidence intervals adjusted for measurement dependency and measurement error of the aggregate estimates. For each tested model, contextual residuals can be stored for later re-use.

Usage

```
ResampleMLSpawAggregate(individual.level.data,
  context.id,
  formula,
  aggregates,
  precise.data = NULL,
  confidence.intervals = c(0.95),
  individual.sample.seed = NULL,
  verbose = TRUE,
  ...)
```

Arguments

<code>individual.level.data</code>	A data.frame containing the individual level predictor and outcome variables. The data must be numeric (logical, integer, or plain numeric). There must not be any missing values, NaNs, NULLs or NAs.
<code>context.id</code>	The name of the context ID variable. This variable allows matching contextual units from different data sets (<code>precise.data</code> , <code>individual.level.data</code>). Values can be of any type but need to be defined consistently across data sets.
<code>formula</code>	Formula description of the model. The formula is handed down to <code>lmer</code> function of the package <code>lme4</code> . Refer to the documentation of <code>lmer</code> for details.
<code>aggregates</code>	A <code>SpawAggregateOutput-class</code> object generated using <code>SpawAggregate</code> , see examples.
<code>precise.data</code>	A data.frame containing the precise contextual data. There must not be any missing values, NaNs, NULLs or NAs. By default set to <code>NULL</code> .
<code>confidence.intervals</code>	vector of confidence intervals. Defaults to <code>c(.95)</code> which corresponds to 95 %.
<code>individual.sample.seed</code>	Seed used to generate the random sampling of the individual data. Is one of three things <ol style="list-style-type: none"> 1. <code>NULL</code>, in which case whatever the current random seed is is used 2. an integer, which will be used to set the random seed. This allows reproducible random samples 3. a saved <code>.Random.seed</code> which allows reproducible random samples as well. The reason why both 2 and 3 are present is because <code>.Random.seed</code> can be saved a posteriori.

Defaults to NULL.

verbose if TRUE, progress information and an estimate of the remaining computing time is shown at the screen. Defaults to TRUE

... All additional named arguments are handed through to the function `lme:lmer` (for example, `REML=TRUE`).

Value

An object of class `SpawAggregateOutput-class`.

Author(s)

Till Junge, Sandra Penic, Guy Elcheroth

References

Elcheroth, G., Penic, S., Fasel, R., Giudici, F., Glaeser, S., Joye, D., Le Goff, J.-M., Morselli, D., & Spini, D. (2012). Spatially weighted context data: a new approach for modelling the impact of collective experiences. *LIVES Working Papers*, 19.

Examples

```
## Spatially weighted multilevel analysis, with resampled both individual
## level indicators and contextual predictors for aggregation. It may
## include non-resampled precise contextual indicator.
## It is step-3 function
## Predicting collective guilt acceptance.

## Data preparation
## load individual level data, remove collective guilt assignment from the
## data frame, and remove NA's
data(traces_ind)
traces_ind <- traces_ind[,-7]
traces_ind <- na.exclude(traces_ind)

## load contextual indicator for aggregation
data(traces_event)

## load precise contextual indicator
data(homog_census)

## load distance matrix
data(d_geo)

## Step 1: Create spatial weights
geow.100 <- WeightMatrix(d_geo, bandwidth=100)

## Step 2: Create spatially weighted aggregated and precise indicators
```

```

wv.agg.100 <- SpawAggregate(
  contextual.data = traces_event,
  context.id="area.name",
  contextual.names = "w_all",
  contextual.weight.matrices=geow.100,
  nb.resamples=5,
  aggregation.functions="weighted.mean",
  design.weight.names="weight",
  sample.seed=1)

homog.100 <- SpawExact(precise.data=homog_census,
  context.id="area.name",
  contextual.names="Homog_00",
  contextual.weight.matrices=geow.100)

## rename weighted variable names so they reflect the used weighting
## matrix
names(wv.agg.100) <- "w_all.100"
names(homog.100)[2] <- "Homog.100"

## Step 3: Perform ResampleMLSpawAggregate
acc_w_homog_100 <-
ResampleMLSpawAggregate(
  individual.level.data=traces_ind,
  context.id="area.name",
  formula=cg_acc ~ victim_d + comb_d + male + age_1990 + high_school +
  higher_edu + (1|area.name) + w_all.100 + Homog.100,
  aggregates=wv.agg.100,
  precise.data=homog.100)

```

ResampleMLSpawExact *Spatially weighted multilevel analysis with stratified resampling for exact context data*

Description

This step-3 function performs multilevel analyses with spatially weighted context data based on precise macro-level measures. The context data created in step 2 are combined with additional individual outcome and (optional) predictor variables, to test a user-defined model. An ad hoc stratified bootstrap resampling procedure generates robust point estimates for regression coefficients and model fit indicators, and computes confidence intervals adjusted for measurement dependency. For each tested model, contextual residuals can be stored for later re-use.

Usage

```

ResampleMLSpawExact(individual.level.data,
  context.id,
  formula,

```

```

precise.data,
confidence.intervals = c(0.95),
nb.resamples = 1000,
individual.sample.seed = NULL,
verbose = TRUE,
...)
```

Arguments

<code>individual.level.data</code>	A data.frame containing the individual level predictor and outcome variables. The data must be numeric (logical, integer, or plain numeric). There must not be any missing values, NaNs, NULLs or NAs.
<code>context.id</code>	This variable allows matching contextual units from different data sets (<code>precise.data</code> , <code>individual.level.data</code>). Values can be of any type but need to be defined consistently across data sets.
<code>formula</code>	Formula description of the model. The formula is handed down to <code>lmer</code> function of the package <code>lme4</code> . Refer to the documentation of <code>lmer</code> for details.
<code>precise.data</code>	A data.frame containing the precise contextual data. There must not be any missing values, NaNs, NULLs or NAs. By default set to NULL.
<code>confidence.intervals</code>	vector of confidence intervals. Defaults to <code>c(.95)</code> which corresponds to 95 %.
<code>nb.resamples</code>	number of resamples to be evaluated. By default set to 1000.
<code>individual.sample.seed</code>	Seed used to generate the random sampling of the individual data. Is one of three things <ol style="list-style-type: none"> 1. NULL, in which case whatever the current random seed is is used 2. an integer, which will be used to set the random seed. This allows reproducible random samples 3. a saved <code>.Random.seed</code> which allows reproducible random samples as well. The reason why both 2 and 3 are present is because <code>.Random.seed</code> can be saved a posteriori. Defaults to NULL.
<code>verbose</code>	if TRUE, progress information and an estimate of the remaining computing time is shown at the screen. Defaults to TRUE
<code>...</code>	All additional named arguments are handed through to the function <code>lme:lmer</code> (for example, <code>REML=TRUE</code>).

Value

An object of class `SpawAggregateOutput-class`.

Author(s)

Till Junge, Sandra Penic, Guy Elcheroth

References

Elcheroth, G., Penic, S., Fasel, R., Giudici, F., Glaeser, S., Joye, D., Le Goff, J.-M., Morselli, D., & Spini, D. (2012). Spatially weighted context data: a new approach for modelling the impact of collective experiences. *LIVES Working Papers*, 19.

Examples

```
# Spatially weighted multilevel analysis, with resampled individual
# level indicators and precise contextual indicator.

## It is step-2 function

## Data preparation

## load individual level data, remove collective guilt assignment from the
## data frame, and remove NA's
data(traces_ind)
traces_ind <- traces_ind[,-7]
traces_ind <- na.exclude(traces_ind)

## load precise contextual indicator
data(homog_census)

## load distance matrix
data(d_geo)

## Step 1: Create spatial weights
geow.100 <- WeightMatrix(d_geo, bandwidth=100)

## Step 2: Create spatially weighted precise contextual indicator
homog.100 <- SpawExact(precise.data=homog_census,
                      context.id="area.name",
                      contextual.names="Homog_00",
                      contextual.weight.matrices=geow.100)
## rename weighted variable names so they reflect the used weighting
## matrix
names(homog.100)[2] <- "Homog.100"

## Step 3: Perform ResampleMLSpawExact
acc_homog100 <-
  ResampleMLSpawExact(
    individual.level.data=traces_ind,
    context.id="area.name",
    formula=cg_acc ~ victim_d + comb_d + male + age_1990 + high_school +
    higher_edu + Homog.100 + (1|area.name), precise.data=homog.100,
    nb.resamples=10)
```

ResampleMLSpawOutput-class

Class "ResampleMLSpawOutput"

Description

Output object of the function ResampleMLSpaw.

Objects from the Class

Objects can be created by calls of the form `new("ResampleMLSpawOutput", ...)`.

Slots

`individual.sample.seed`: Object of class "integer"

`fixed`: Object of class "data.frame"

`random.var`: Object of class "data.frame"

`model.fit`: Object of class "data.frame"

`ranefs`: Object of class "matrix"

`nb.resamples`: Object of class "integer"

`betas`: Object of class "data.frame"

Methods

print signature(x = "ResampleMLSpawOutput")

show signature(object = "ResampleMLSpawOutput")

Author(s)

Till Junge

Examples

```
## Perform ResampleMLSpawExact

## Data preparation (see ResampleMLSpawExact)

data(traces_ind)
traces_ind <- traces_ind[,-7]
traces_ind <- na.exclude(traces_ind)
data(homog_census)
data(d_geo)

## Step 1: Create spatial weights
geow.100 <- WeightMatrix(d_geo, bandwidth=100)

## Step 2: Create spatially weighted precise contextual indicator
homog.100 <- SpawExact(precise.data=homog_census,
                      context.id="area.name",
                      contextual.names="Homog_00",
                      contextual.weight.matrices=geow.100)
## rename weighted variable names so they reflect the used weighting
## matrix
names(homog.100)[2] <- "Homog.100"
```

```

## Step 3: Perform ResampleMLSpawExact
acc_homog100 <-
  ResampleMLSpawExact(
    individual.level.data=traces_ind,
    context.id="area.name",
    formula=cg_acc ~ victim_d + comb_d + male + age_1990 + high_school +
    higher_edu + Homog.100 + (1|area.name), precise.data=homog.100,
    nb.resamples=10)

## acc_homog100 is an object of class ResampleMLSpawOutput
class(acc_homog100)

## to assess standardized fixed effects coefficients
acc_homog100@betas

## to assess non-standardized fixed effects coefficients
acc_homog100@fixed

## to assess only median of non-standardized fixed effects coefficients
## Not run:
acc_homog100@fixed["50%"]
## End(Not run)

## to assess random effects
acc_homog100@random.var

```

SpawAggregate

Create and describe spatially weighted survey estimates for contextual indicators

Description

This step-2 function should be used to create spatially weighted context data when contextual indicators are aggregate estimates derived from available micro-level survey data. Aggregation functions can be user-defined or chosen among predefined options, which include indicators of central tendency (e.g., spatially weighted mean), dispersion (e.g., spatially weighted standard deviation) and inter-group variability (e.g., spatially weighted Gini coefficient for group inequality). Point estimates can be adjusted by user-defined design weights. Descriptive confidence intervals are computed by way of an ad hoc bootstrap resampling procedure.

Usage

```

SpawAggregate(contextual.data,
  context.id,
  contextual.names,
  contextual.weight.matrices=NULL,
  nb.resamples = 1000,
  aggregation.functions = "weighted.mean",

```

```

confidence.intervals = 0.95,
design.weight.names = NULL,
sample.seed = NULL,
additional.args = NULL,
verbose = TRUE)

```

Arguments

- `contextual.data` A data.frame consisting of micro-level data to generate contextual indicators by aggregation and containing a column named `context.id` with the context ID variable. This is an individual level dataframe. The data must be numeric (logical, integer, or plain numeric). There must not be any missing values, NaNs, NULLs or NAs.
- `context.id` The name of the context ID variable. This variable allows matching contextual units from different data sets (`contextual.data`, `contextual.weight.matrix`). Values can be of any type but need to be defined consistently across data sets.
- `contextual.names` A list of names of contextual variables to be aggregated.
- `contextual.weight.matrices` A list of weights to be applied to each variable specified in `contextual.names`. A weight may be a weights matrix as, for instance computed by `WeightMatrix`, or NULL, in which case the corresponding contextual variable is not weighted. If only one weight is defined (instead of a list) it is applied to all contextual variables. Defaults to NULL, which means that none of contextual indicators are weighted.
- `nb.resamples` A number of resamples to be evaluated. By default set to 1000.
- `aggregation.functions` A list of aggregation functions. Functions take either
- 1 argument in which case the corresponding individual design weight is NULL,
 - 2 arguments in which case the second argument is taken from the corresponding individual design weight. Defaults to "mean".
- `confidence.intervals` A vector of confidence intervals. Defaults to 0.95 which corresponds to 95 %.
- `design.weight.names` a list of optional design weights at the individual level used for aggregation (for example, for a weighted mean). List must have same length as `contextual.names`. May contain NULLs for variables which should not be weighted at the individual level. If only one design.weight is defined (instead of a list) it is applied to all contextual variables. By default set to NULL.
- `sample.seed` Is one of three things
1. NULL, in which case whatever the current random seed is used
 2. an integer, which will be used to set the random seed. This allows reproducible random samples


```

    aggregation.functions="weighted.mean",
    design.weight.names="weight",
    nb.resamples=5)

## To create non-weighted and spatially weighted (by geow.50) risk of war victimization

ww.nw.g50 <- SpawAggregate(contextual.data=traces_event,
    context.id="area.name",
    contextual.names=c("w_all", "w_all"),
    contextual.weight.matrices=list(NULL, geow.50),
    aggregation.functions="weighted.mean",
    design.weight.names="weight",
    nb.resamples=5)

```

SpawAggregateOutput-class

Class "SpawAggregateOutput"

Description

Output object of the function SpawAggregate.

Objects from the Class

Objects can be created by calls of the form `new("SpawAggregateOutput", ...)`.

Slots

seed: Object of class "ANY" ~~

contains the random seed which was used in SpawAggregate. Use this seed if you want to reproduce the same resamples.

aggregated.samples: Object of class "list" ~~

A list of matrices of dimension $n \times m$ where n is the number of contextual units and m is the number of resamples containing the aggregated context data for each resample.

frames: Object of class "list" ~~

A list of data.frames. One data.frame per contextual name, each containing the mean estimate, standard deviation, median and confidence intervals of the corresponding contextual variable by contextual unit.

Methods

merge signature(`x = "SpawAggregateOutput"`, `y = "SpawAggregateOutput"`):
Merges SpawAggregateOutput objects. This is useful if contextual data is being aggregated from different individual level data bases.

[signature(`x = "SpawAggregateOutput"`, `i = "ANY"`, `j = "ANY"`, `drop = "ANY"`):
Allows to extract the dataframe corresponding to a bootstrap sample.

getSample signature(object = "SpawAggregateOutput"): Also allows to extract the dataframe corresponding to a bootstrap sample.

length signature(x = "SpawAggregateOutput"): Returns the number of bootstrap samples contained in the object.

names signature(x = "SpawAggregateOutput"): Returns the names of aggregated variables in the object

names<- signature(x = "SpawAggregateOutput"): Allows to assign names to the aggregates in the object

names<- signature(x = "SpawAggregateOutput", NA = "ANY"): Allows to change names to the aggregated variables in the object.

show signature(object = "SpawAggregateOutput"): Prints a meaningful short description of the object.

Author(s)

Till Junge, Sandra Penic, Guy Elcheroth

See Also

is generated by [SpawAggregate](#)

Examples

```
## load distance matrix
data(d_geo)
## create a weight matrix
geow.50 <- WeightMatrix(d_geo, bandwidth=50)
## load contextual data
data(traces_event)

## aggregate data

wv.int <- SpawAggregate(contextual.data=traces_event,
                       context.id="area.name",
                       contextual.names=c("w_all", "w_all"),
                       contextual.weight.matrices=list(NULL, geow.50),
                       aggregation.functions="weighted.mean",
                       design.weight.names="weight",
                       nb.resamples=4)

## check the number of samples
length(wv.int)

## check the second sample
head(wv.int[2])

## change the names of the weighted indicators to more significant ones
names(wv.int) <- c("w_all.0", "w_all.50")

showClass("SpawAggregateOutput")
```

`SpawExact`*Create and describe spatially weighted precise contextual indicators*

Description

This step-2 function should be used to create spatially weighted context data when contextual indicators are directly available as precise macro-level measures (e.g., population values based on census or registry data, or precise geographic measures). The generated context data can be adjusted by user-defined population weights.

Usage

```
SpawExact(precise.data,  
          context.id,  
          contextual.names,  
          contextual.weight.matrices,  
          population.weight.names = NULL)
```

Arguments

<code>precise.data</code>	A data.frame consisting of precise contextual indicators and a context ID variable
<code>context.id</code>	The name of the context ID variable. This variable allows matching contextual units from different data sets (<code>precise.data</code> , <code>contextual.weight.matrix</code>). Values can be of any type but need to be defined consistently across data sets.
<code>contextual.names</code>	A list of column names of contextual variables to be weighted
<code>contextual.weight.matrices</code>	A list of square weight matrices of dimension $n \times n$, where n is the number of contextual units to be applied to the variable specified in <code>contextual.names</code> , respectively.
<code>population.weight.names</code>	A list of column names to be used as population weights. By default NULL which means that no population weighting is performed. If only one name is provided, the corresponding column is used as a weight for all contextual names

Value

A data.frame containing the spatially weighted `contextual.names`.

Author(s)

Till Junge, Sandra Penic, Guy Elcheroth

References

Elcheroth, G., Penic, S., Fasel, R., Giudici, F., Glaeser, S., Joye, D., Le Goff, J.-M., Morselli, D., & Spini, D. (2012). Spatially weighted context data: a new approach for modelling the impact of collective experiences. *LIVES Working Papers*, 19.

Examples

```
## Creating spatially weighted precise contextual indicator - spatially weighted
## ethnic homogeneity across TRACES areas.
## This is step-2 function

## Step 1: Load distance matrix and create weights

data(d_geo)
geow.50 <- WeightMatrix(d_geo, bandwidth=50)

## Step 2: Load dataframe with precise contextual indicator and
## create spatially weighted indicator

data(homog_census)

homog.g50 <- SpawExact(precise.data=homog_census,
                      context.id="area.name",
                      contextual.names="Homog_00",
                      contextual.weight.matrices=geow.50)
```

traces_event

Dataset traces_event

Description

A dataset with micro-level data to generate contextual indicator of risk of war victimization by aggregation. The dataset consists of 243110 cases and three variables. The cases are individuals by three-month time periods.

Usage

```
traces_event
```

Format

The dataframe contains the following columns:

area - context ID variable, ranges from 1 to 80

w_all - experience of a war event (0 - no, 1 - yes)

weight - design weights

Source

TRACES dataset. Full TRACES datasets can be accessed through the Data and Research Information Services from the Swiss Foundation for Research in the Social Sciences (www.unil.ch/daris).

References

Elcheroth, G., Penic, S., Fasel, R., Giudici, F., Glaeser, S., Joye, D., Le Goff, J.-M., Morselli, D., & Spini, D. (2012). Spatially weighted context data: a new approach for modelling the impact of collective experiences. *LIVES Working Papers*, 19.

Spini, D., Elcheroth, G., & Fasel, R. (2011). TRACES: Methodological and technical report. *LIVES Working Papers*, 4.

Examples

```
data(traces_event)
```

traces_ind

Dataset traces_ind

Description

Individual level data set, with individual level predictor and outcome variables. The traces_ind dataframe includes responses of 2254 respondents on 12 variables.

Usage

```
traces_ind
```

Format

This dataframe contains the following columns:

area - context ID variable, ranges from 1 to 80

label - labels of areas

entity - political entities of former Yugoslavia in 2006 (1 - Slovenia, 2 - Croatia, 3 - Bosnia, 4 - Serbia, 5 - Vojvodina, 6 - Montenegro, 7 - Kosovo, 8 - Macedonia)

victim_d - personal experience of war victimization (0-no, 1-yes)

comb_d - participation in combats (0-no, 1-yes)

male - gender (0-woman, 1-man)

cg_acc - composite results on 5 items of the Collective guilt acceptance scale, ranges from 1 to 7

cg_ass - composite results on 5 items of the Collective guilt assignment scale, ranges from 1 to 7

age_1990 - age of respondents, in years

elem_school - elementary school as a highest level of education (0-no, 1-yes)

high_school - high school as a highest level of education (0-no, 1-yes)

higher_edu - achieved higher level education (0-no, 1-yes)

Source

TRACES. Full TRACES datasets can be accessed through the Data and Research Information Services from the Swiss Foundation for Research in the Social Sciences (www.unil.ch/daris).

References

Spini, D., Elcheroth, G., & Fasel, R. (2011). TRACES: Methodological and technical report. *LIVES Working Papers*, 4.

Examples

```
data(traces_ind)
```

WeightMatrix	<i>Create spatial weighting matrices</i>
--------------	--

Description

This step-1 function creates a matrix of spatial weights on the basis of a user-defined distance matrix, Kernel function, and bandwidth value. The distance matrix needs to specify a value for each of the possible $n \times n$ binomials that correspond to n contextual units. It can be either symmetric or asymmetric. In principle, its diagonal, corresponding to the distance of each unit with itself, should be composed of zero values. A Kernel function proposed by default generates spatial weights that tend toward 1 for distances substantially lower than the bandwidth value, toward 0 for distances substantially higher than the bandwidth value and toward 0.5 for distances approaching the bandwidth value.

Usage

```
WeightMatrix(distance.matrix, bandwidth, kernel = NULL,
             moran = FALSE)
```

Arguments

`distance.matrix` square matrix of dimension $n \times n$, where n is the number of contextual units.

`bandwidth` scalar numeric value specifying the bandwidth h

`kernel` function applied to the distance matrix. By default NULL, in which case the kernel function

$$w_{ij} = f(d, h) = \left(\frac{1}{2}\right)^{d_{ij}^2/h^2}$$

is used, where w_{ij} , d_{ij} , h are elements of the weight matrix \mathbf{W} , of the distance matrix \mathbf{D} and the bandwidth h . User-supplied kernel functions have to take 2 arguments and return a matrix of the same dimension as the first argument.

`moran` a logical value specifying whether the proximity weights matrix should have zeros in the diagonal. By default set to FALSE.

Value

A weights matrix of the same dimension as `distance.matrix`.

Author(s)

Till Junge, Sandra Penic, Guy Elcheroth

References

Elcheroth, G., Penic, S., Fasel, R., Giudici, F., Glaeser, S., Joye, D., Le Goff, J.-M., Morselli, D., & Spini, D. (2012). Spatially weighted context data: a new approach for modelling the impact of collective experiences. *LIVES Working Papers*, 19.

Examples

```
# creating geographical proximity weight, with bandwidth h=50  
  
data(d_geo)  
geow_50 <- WeightMatrix(distance.matrix=d_geo, bandwidth=50)
```

Index

- *Topic **classes**
 - MLSpawExactOutput-class, 16
 - ResampleMLSpawOutput-class, 25
 - SpawAggregateOutput-class, 30
- *Topic **datasets**
 - CH1903, 8
 - GPS, 13
- *Topic **spacom**
 - WeightMatrix, 35
- *Topic **spatial**
 - aggregation functions, 5
 - CH1903, 8
 - ExploreMLSpawExact, 10
 - MLSpawExact, 14
 - MLSpawExactOutput-class, 16
 - ResampleMLSpawAggregate, 20
 - ResampleMLSpawExact, 23
 - ResampleMLSpawOutput-class, 25
 - SpawAggregate, 27
 - SpawAggregateOutput-class, 30
 - SpawExact, 32
 - WeightMatrix, 35
- [, SpawAggregateOutput, ANY, ANY, ANY-method
(SpawAggregateOutput-class), 30
- aggregation functions, 5
- AIC, MLSpawExactOutput-method
(MLSpawExactOutput-class), 16
- BIC, MLSpawExactOutput-method
(MLSpawExactOutput-class), 16
- CH1903, 8
- d_ethnic, 8
- d_geo, 9
- d_ident, 9
- d_migr, 10
- ExploreMLSpawAggregate, 5
- ExploreMLSpawAggregate
(ExploreMLSpawExact), 10
- ExploreMLSpawExact, 3, 10
- fixef, MLSpawExactOutput-method
(MLSpawExactOutput-class), 16
- getSample, SpawAggregateOutput-method
(SpawAggregateOutput-class), 30
- GPS, 13
- homog_census, 14
- length, SpawAggregateOutput-method
(SpawAggregateOutput-class), 30
- merge, SpawAggregateOutput, SpawAggregateOutput-method
(SpawAggregateOutput-class), 30
- merMod, 17
- MLSpawExact, 3, 14
- MLSpawExactOutput-class, 16
- MLSpawResidMoran, 3, 18
- names, SpawAggregateOutput-method
(SpawAggregateOutput-class), 30
- names<- , SpawAggregateOutput, ANY-method
(SpawAggregateOutput-class), 30
- names<- , SpawAggregateOutput-method
(SpawAggregateOutput-class), 30
- print, MLSpawExactOutput-method
(MLSpawExactOutput-class), 16
- print, ResampleMLSpawOutput-method
(ResampleMLSpawOutput-class),
25
- ranef, MLSpawExactOutput-method
(MLSpawExactOutput-class), 16
- ResampleMLSpawAggregate, 3, 20
- ResampleMLSpawExact, 3, 23
- ResampleMLSpawOutput-class, 25

show, MLSpawExactOutput-method
(MLSpawExactOutput-class), 16

show, ResampleMLSpawOutput-method
(ResampleMLSpawOutput-class),
25

show, SpawAggregateOutput-method
(SpawAggregateOutput-class), 30

spacom (spacom-package), 2

spacom-package, 2

SpawAggregate, 3, 5, 21, 27, 31

SpawAggregateOutput-class, 30

SpawExact, 3, 32

summary, MLSpawExactOutput-method
(MLSpawExactOutput-class), 16

traces_event, 33

traces_ind, 34

VarCorr, MLSpawExactOutput-method
(MLSpawExactOutput-class), 16

WeightMatrix, 3, 28, 35

wt.Atkinson (aggregation functions), 5

wt.gini (aggregation functions), 5

wt.mean (aggregation functions), 5

wt.RS (aggregation functions), 5

wt.sd (aggregation functions), 5

wt.Theil (aggregation functions), 5

wt.var (aggregation functions), 5