Package 'socialmixr'

October 18, 2024

Title Social Mixing Matrices for Infectious Disease Modelling

Version 0.4.0

Description Provides methods for sampling contact matrices from diary data for use in infectious disease modelling, as discussed in Mossong et al. (2008) <doi:10.1371/journal.pmed.0050074>.

License MIT + file LICENSE

Depends R (>= 3.5.0)

Imports countrycode, curl, data.table, grDevices, httr, jsonlite, lubridate, memoise, oai, wpp2017, xml2

Suggests ggplot2, here, knitr, purrr, reshape2, rmarkdown, roxyglobals (>= 1.0.0), testthat

VignetteBuilder knitr

Encoding UTF-8

LazyData true

NeedsCompilation no

RoxygenNote 7.3.2

URL https://github.com/epiforecasts/socialmixr, https://epiforecasts.io/socialmixr/

BugReports https://github.com/epiforecasts/socialmixr/issues

Author Sebastian Funk [aut, cre], Lander Willem [aut], Hugo Gruson [aut], Maria Bekker-Nielsen Dunbar [ctb], Carl A. B. Pearson [ctb], Sam Clifford [ctb], Christopher Jarvis [ctb], Alexis Robert [ctb], Niel Hens [ctb], Pietro Coletti [col, dtm]

Maintainer Sebastian Funk <sebastian.funk@lshtm.ac.uk>

Repository CRAN

Date/Publication 2024-10-18 15:20:01 UTC

Contents

`check` *Check contact survey data*

Description

Checks that a survey fulfills all the requirements to work with the 'contact_matrix' function

Usage

```
## S3 method for class 'survey'
check(
  x,
  id.column = "part_id",
  participant.age.column = "part_age",
  country.column = "country",
  year.column = "year",
  contact.age.column = "cnt_age",
  ...
)
```

Arguments

x	A survey() object
id.column	the column in both the participants and contacts data frames that links contacts to participants
participant.age.column	the column in the participants data frame containing participants' age; if this does not exist, at least columns "..._exact", "..._est_min" and "..._est_max" must (see the estimated.participant.age option in contact_matrix())
country.column	the column in the participants data frame containing the country in which the participant was queried
year.column	the column in the participants data frame containing the year in which the participant was queried
contact.age.column	the column in the contacts data frame containing the age of contacts; if this does not exist, at least columns "..._exact", "..._est_min" and "..._est_max" must (see the estimated.contact.age option in contact_matrix())
...	ignored

Value

invisibly returns a character vector of the relevant columns

Examples

```
data(polymod)
check(polymod)
```

---

clean *Clean contact survey data*

Description

Cleans survey data to work with the 'contact_matrix' function

Usage

```
## S3 method for class 'survey'
clean(x, country.column = "country", participant.age.column = "part_age", ...)
```

Arguments

x	A survey() object
country.column	the name of the country in which the survey participant was interviewed
participant.age.column	the column in x$participants containing participants' age
...	ignored

Value

a cleaned survey in the correct format

Examples

```
data(polymod)
cleaned <- clean(polymod) # not really necessary as the 'polymod' data set has already been cleaned
```

`contact_matrix` *Generate a contact matrix from diary survey data*

Description

Samples a contact survey

Usage

```
contact_matrix(
  survey,
  countries = NULL,
  survey.pop,
  age.limits,
  filter,
  counts = FALSE,
  symmetric = FALSE,
  split = FALSE,
  sample.participants = FALSE,
  estimated.participant.age = c("mean", "sample", "missing"),
  estimated.contact.age = c("mean", "sample", "missing"),
  missing.participant.age = c("remove", "keep"),
  missing.contact.age = c("remove", "sample", "keep", "ignore"),
  weights = NULL,
  weigh.dayofweek = FALSE,
  weigh.age = FALSE,
  weight.threshold = NA,
  sample.all.age.groups = FALSE,
  return.part.weights = FALSE,
  return.demography = NA,
  per.capita = FALSE,
  ...
)
```

Arguments

`survey`	a `survey()` object
`countries`	limit to one or more countries; if not given, will use all countries in the survey; these can be given as country names or 2-letter (ISO Alpha-2) country codes

`survey.pop` survey population – either a data frame with columns 'lower.age.limit' and 'population', or a character vector giving the name(s) of a country or countries from the list that can be obtained via `wpp_countries`; if not given, will use the country populations from the chosen countries, or all countries in the survey if `countries` is not given

`age.limits` lower limits of the age groups over which to construct the matrix

`filter` any filters to apply to the data, given as list of the form (column=filter_value) - only contacts that have 'filter_value' in 'column' will be considered. If multiple filters are given, they are all applied independently and in the sequence given.

`counts` whether to return counts (instead of means)

`symmetric` whether to make matrix symmetric, such that $c_{ij}N_i = c_{ji}N_j$.

`split` whether to split the contact matrix into the mean number of contacts, in each age group (split further into the product of the mean number of contacts across the whole population (`mean.contacts`), a normalisation constant (`normalisation`) and age-specific variation in contacts (`contacts`)), multiplied with an assortativity matrix (`assortativity`) and a population multiplier (`demograpy`). For more detail on this, see the "Getting Started" vignette.

`sample.participants` whether to sample participants randomly (with replacement); done multiple times this can be used to assess uncertainty in the generated contact matrices. See the "Bootstrapping" section in the vignette for how to do this..

`estimated.participant.age` if set to "mean" (default), people whose ages are given as a range (in columns named "..._est_min" and "..._est_max") but not exactly (in a column named "..._exact") will have their age set to the mid-point of the range; if set to "sample", the age will be sampled from the range; if set to "missing", age ranges will be treated as missing

`estimated.contact.age` if set to "mean" (default), contacts whose ages are given as a range (in columns named "..._est_min" and "..._est_max") but not exactly (in a column named "..._exact") will have their age set to the mid-point of the range; if set to "sample", the age will be sampled from the range; if set to "missing", age ranges will be treated as missing

`missing.participant.age` if set to "remove" (default), participants without age information are removed; if set to "keep", participants with missing age are kept and treated as a separate age group

`missing.contact.age` if set to "remove" (default), participants that have contacts without age information are removed; if set to "sample", contacts without age information are sampled from all the contacts of participants of the same age group; if set to "keep", contacts with missing age are kept and treated as a separate age group; if set to "ignore", contact with missing age are ignored in the contact analysis

`weights` column names(s) of the participant data of the `survey()` object with user-specified weights (default = empty vector)

`weigh.dayofweek`
: whether to weigh social contacts data by the day of the week (weight (5/7 / N_week / N) for weekdays and (2/7 / N_weekend / N) for weekends)

`weigh.age`
: whether to weigh social contacts data by the age of the participants (vs. the populations' age distribution)

`weight.threshold`
: threshold value for the standardized weights before running an additional standardisation (default 'NA' = no cutoff)

`sample.all.age.groups`
: what to do if sampling participants (with `sample.participants = TRUE`) fails to sample participants from one or more age groups; if FALSE (default), corresponding rows will be set to NA, if TRUE the sample will be discarded and a new one taken instead

`return.part.weights`
: boolean to return the participant weights

`return.demography`
: boolean to explicitly return demography data that corresponds to the survey data (default 'NA' = if demography data is requested by other function parameters)

`per.capita`
: whether to return a matrix with contact rates per capita (default is FALSE and not possible if 'counts=TRUE' or 'split=TRUE')

`...`
: further arguments to pass to `get_survey()`, `check()` and `pop_age()` (especially column names)

Value

a contact matrix, and the underlying demography of the surveyed population

Author(s)

Sebastian Funk

Examples

```
data(polymod)
contact_matrix(polymod, countries = "United Kingdom", age.limits = c(0, 1, 5, 15))
```

---

download_survey *Download a survey from its Zenodo repository*

Description

Downloads survey data

Usage

```
download_survey(survey, dir = NULL, sleep = 1)
```

Arguments

survey	a URL (see list_surveys())
dir	a directory to save the files to; if not given, will save to a temporary directory
sleep	time to sleep between requests to avoid overloading the server (passed on to Sys.sleep)

Value

a vector of filenames that can be used with load_survey

Examples

```
## Not run:
list_surveys()
peru_survey <- download_survey("https://doi.org/10.5281/zenodo.1095664")

## End(Not run)
```

---

get_citation *Citation for a survey*

Description

Gets a full citation for a survey().

Usage

```
get_citation(x)
```

Arguments

x	a character vector of surveys to cite

Value

citation as bibentry

Examples

```
data(polymod)
citation <- get_citation(polymod)
print(citation)
print(citation, style = "bibtex")
```

get_survey *Get a survey, either from its Zenodo repository, a set of files, or a survey variable*

Description

Downloads survey data, or extracts them from files, and returns a clean data set. If a survey URL is accessed multiple times, the data will be cached (unless `clear_cache` is set to `TRUE`) to avoid repeated downloads.

Usage

```
get_survey(survey, clear_cache = FALSE, ...)
```

Arguments

`survey`	a DOI or url to get the survey from, or a `survey()` object (in which case only cleaning is done).
`clear_cache`	logical, whether to clear the cache before downloading the survey; by default, the cache is not cleared and so multiple calls of this function to access the same survey will not result in repeated downloads
`...`	options for `clean()`, which is called at the end of this

Details

If survey objects are used repeatedly the downloaded files can be saved and reloaded between sessions then survey objects can be saved/loaded using `base::saveRDS()` and `base::readRDS()`, or via the individual survey files that can be downloaded using `download_survey()` and subsequently loaded using `load_survey()`.

Value

a survey in the correct format

Examples

```
## Not run:
list_surveys()
peru_survey <- get_survey("https://doi.org/10.5281/zenodo.1095664")

## End(Not run)
```

is_doi *Checks if a character string is a DOI*

Description

Checks if a character string is a DOI

Usage

```
is_doi(x)
```

Arguments

x	Character vector; the string or strings to check

Value

Logical; TRUE if x is a DOI, FALSE otherwise

Author(s)

Sebastian Funk

limits_to_agegroups *Convert lower age limits to age groups.*

Description

Mostly used for plot labelling

Usage

```
limits_to_agegroups(
  x,
  limits = sort(unique(x)),
  notation = c("dashes", "brackets")
)
```

Arguments

x	age limits to transform
limits	lower age limits; if not given, will use all limits in x
notation	whether to use bracket notation, e.g. [0,4) or dash notation, e.g. 0-4)

Value

Age groups as specified in `notation`

Examples

```
limits_to_agegroups(c(0, 5, 10))
```

`list_surveys` *List all surveys available for download*

Description

List all surveys available for download

Usage

```
list_surveys(clear_cache = FALSE)
```

Arguments

`clear_cache`	logical, whether to clear the cache before downloading the survey; by default, the cache is not cleared and so multiple calls of this function to access the same survey will not result in repeated downloads

Value

character vector of surveys

Examples

```
## Not run:
list_surveys()

## End(Not run)
```

load_survey *Load a survey from local files*

Description

Loads a survey from a local file system. Tables are expected as csv files, and a reference (if present) as JSON.

Usage

```
load_survey(files, ...)
```

Arguments

files	a vector of file names as returned by download_survey()
...	options for clean(), which is called at the end of this

Value

a survey in the correct format

Examples

```
## Not run:
list_surveys()
peru_files <- download_survey("https://doi.org/10.5281/zenodo.1095664")
peru_survey <- load_survey(peru_files)

## End(Not run)
```

matrix_plot *Draws an image plot of a contact matrix with a legend strip and the numeric values in the cells.*

Description

This function combines the R image.plot function with numeric contact rates in the matrix cells.

Usage

```
matrix_plot(
  mij,
  min.legend = 0,
  max.legend = NA,
  num.digits = 2,
  num.colors = 50,
```

```
  main,
  xlab,
  ylab,
  legend.width,
  legend.mar,
  legend.shrink,
  cex.lab,
  cex.axis,
  cex.text,
  color.palette = heat.colors
)
```

Arguments

mij	a contact matrix containing contact rates between participants of age i (rows) with contacts of age j (columns). This is the default matrix format of contact_matrix().
min.legend	the color scale minimum (default = 0). Set to NA to use the minimum value of mij.
max.legend	the color scale maximum (default = NA). Set to NA to use the maximum value of mij.
num.digits	the number of digits when rounding the contact rates (default = 2). Use NA to disable this.
num.colors	the number of color breaks (default = 50)
main	the figure title
xlab	a title for the x axis (default: "Age group (years)")
ylab	a title for the y axis (default: "Contact age group (years)")
legend.width	width of the legend strip in characters. Default is 1.
legend.mar	width in characters of legend margin. Default is 5.1.
legend.shrink	amount to shrink the size of legend relative to the full height or width of the plot. Default is 0.9.
cex.lab	size of the x and y labels (default: 1.2)
cex.axis	size of the axis labels (default: 0.8)
cex.text	size of the numeric values in the matrix (default: 1)
color.palette	the color palette to use (default: heat.colors()). Other examples are topo.colors(), terrain.colors() and hcl.colors(). User-defined functions are also possible if they take the number of colors to be in the palette as function argument.

Details

This is a function using basic R graphics to visualise a social contact matrix.

Author(s)

Lander Willem

Examples

```
## Not run:
data(polymod)
mij <- contact_matrix(polymod, countries = "United Kingdom", age.limits = c(0, 18, 65))$matrix
matrix_plot(mij)

## End(Not run)
```

`polymod` *Social contact data from 8 European countries*

Description

A dataset containing social mixing diary data from 8 European countries: Belgium, Germany, Finland, Great Britain, Italy, Luxembourg, The Netherlands and Poland. The Data are fully described in Mossong J, Hens N, Jit M, Beutels P, Auranen K, Mikolajczyk R, et al. (2008) Social Contacts and Mixing Patterns Relevant to the Spread of Infectious Diseases. PLoS Med 5(3): e74.

Usage

```
polymod
```

Format

A list of two data frames:

participants the study participant, with age, country, year and day of the week (starting with 1 = Monday)

contacts reported contacts of the study participants. The variable phys_contact has two levels (1 denotes physical contact while 2 denotes non-physical contact), duration_multi has five levels (1 is less than 5 minutes while 5 is more than 4 hours, increasing in the order found in Figure 1 in Mossong et al.), and frequency_multi has five levels (1 is daily, 2 is weekly, 3 is monthly, 4 is less often, and 5 is first time) All other variables are described on the Zenodo repository of the data, available at doi:10.5281/zenodo.1043437

Source

doi:10.1371/journal.pmed.0050074

pop_age *Change age groups in population data*

Description

This changes population data to have age groups with the given age.limits, extrapolating linearly between age groups (if more are requested than available) and summing populations (if fewer are requested than available)

Usage

```
pop_age(
  pop,
  age.limits,
  pop.age.column = "lower.age.limit",
  pop.column = "population",
  ...
)
```

Arguments

pop	a data frame with columns indicating lower age limits and population sizes (see 'age.column' and 'pop.column')
age.limits	lower age limits of age groups to extract
pop.age.column	column in the 'pop' data frame indicating the lower age group limit
pop.column	column in the 'pop' data frame indicating the population size
...	ignored

Value

data frame of age-specific population data

Examples

```
ages_it_2015 <- wpp_age("Italy", 2015)

# Modify the age data.frame to get age groups of 10 years instead of 5
pop_age(ages_it_2015, age.limit = seq(0, 100, by = 10))

# The function will also automatically interpolate if necessary
pop_age(ages_it_2015, age.limit = c(0, 18, 40, 65))
```

reduce_agegroups *Reduce the number of age groups given a broader set of limits*

Description

Operates on lower limits

Usage

```
reduce_agegroups(x, limits)
```

Arguments

x	vector of limits
limits	new limits

Value

vector with the new age groups

Examples

```
reduce_agegroups(seq_len(20), c(0, 5, 10))
```

survey *Contact survey*

Description

A survey object contains the results of a contact survey. In particular, it contains two data frames called participants and contacts that are linked by a column specified as id.column

Usage

```
survey(participants, contacts, reference = NULL)
```

Arguments

participants	a data.frame containing information on participants
contacts	a data.frame containing information on contacts
reference	a list containing information needed to reference the survey, in particular it can contain$a "title", "bibtype", "author", "doi", "publisher", "note", "year"

Value

a new survey object

Author(s)

Sebastian Funk

Examples

```
data(polymod)
new_survey <- survey(polymod$participants, polymod$contacts)
```

---

survey_countries *List all countries contained in a survey*

Description

List all countries contained in a survey

Usage

```
survey_countries(survey, country.column = "country", ...)
```

Arguments

survey	a DOI or url to get the survey from, or a survey() object (in which case only cleaning is done).
country.column	column in the survey indicating the country
...	further arguments for get_survey()

Value

list of countries

Examples

```
data(polymod)
survey_countries(polymod)
```

wpp_age *Get age-specific population data according to the World Population Prospects 2017 edition*

Description

This uses data from the wpp2017 package but combines male and female, and converts age groups to lower age limits. If the requested year is not present in the historical data, wpp projections are used.

Usage

```
wpp_age(countries, years)
```

Arguments

countries	countries, will return all if not given
years	years, will return all if not given

Value

data frame of age-specific population data

Examples

```
wpp_age("Italy", c(1990, 2000))
```

wpp_countries *List all countries and regions for which socialmixr has population data*

Description

Uses the World Population Prospects data from the wpp2017 package

Usage

```
wpp_countries()
```

Value

list of countries

Examples

```
wpp_countries()
```

Index