Package: jollofR (via r-universe)

October 11, 2026

Type Package

Title Small Area Population Estimation by Demographics

Version 0.6.5

Description Automatic disaggregation of small-area population estimates by demographic groups (e.g., age, sex, race, marital status, educational level, etc) along with the estimates of uncertainty, using advanced Bayesian statistical modelling approaches based on integrated nested Laplace approximation (INLA) Rue et al. (2009) <doi:10.1111/j.1467-9868.2008.00700.x> and stochastic partial differential equation (SPDE) methods Lindgren et al. (2011) <doi:10.1111/j.1467-9868.2011.00777.x>. The package implements hierarchical Bayesian modeling frameworks for small area estimation as described in Leasure et al. (2020) <doi:10.1073/pnas.1913050117> and Nnanatu et al. (2025) <doi:10.1038/s41467-025-59862-4>.

License MIT + file LICENSE

URL https://github.com/wpgp/jollofR/, https://wpgp.github.io/jollofR/

BugReports https://github.com/wpgp/jollofR/issues

Depends R (>= 4.1.0)

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

Imports terra, raster, ggplot2, dplyr, tidyr, sf, ggpubr, reshape2, utils

Suggests INLA, withr, testthat (>= 3.0.0)

Config/testthat/edition 3

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE, r6 = TRUE)

RoxygenNote 7.3.2

Language en-US

Config/Needs/website rmarkdown

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

Date/Publication 2025-12-15 10:13:39 UTC

RemoteUrl https://github.com/wpgp/jollofr

RemoteRef HEAD

RemoteSha 49cbda30f49661b8e57c2780c43bf6e35fc5f5e8

Contents

boxLine 2
cheesecake 3
cheesepop 4
plotHist 5
plotRast 6
pyramid 7
slices 8
spices 9
splash 10
splash1 11
spray 12
spray1 14
sprinkle 15
sprinkle1 16
toydata 18

Index **20**

`boxLine` *boxLine: Produces two graphs - boxplots of disaggregated population counts across groups and a line plot showing the distribution of the aggregated totals of the disaggregated counts*

Description

This function automatically generates two graphs that are combined together - (a) a boxplot of the distribution of the various groups' disaggregated population counts, and (b) a line graph of the aggregated counts across all groups (e.g., total number of individuals for each group). Here, the input data could come from any of the disaggregation functions within the 'jollofR' package such as 'cheesecake', 'cheesepop', 'slices' & 'spices'.

Usage

```
boxLine(dmat, xlab, ylab)
```

Arguments

dmat	A data frame containing the group-structured disaggregated population estimates which could be observed or from modelled estimates based on any of the functions - cheesecake', 'cheesepop', 'slices','spices', 'spray' , 'sprinkle', 'splash', 'spray', 'sprinkle1', 'splash1', or 'spray1'. considered.
xlab	A user-defined label for the x-axis (e.g., 'Age group').
ylab	A user-defined label for the y-axis (e.g., 'Population count').

Value

A graphic image of two combined graphs - a boxplot and a line plot showing the distribution of the disaggregated population counts across the groups.

Examples

```
if (requireNamespace("INLA", quietly = TRUE)) {
 library(ggplot2)
 data(toydata)
 result <- cheesepop(df = toydata$admin,output_dir = tempdir())
 boxLine(dmat=result$male_age_pop,
         xlab="Age group (years)",
         ylab = "Population Count")
}
```

---

cheesecake *cheesecake: Population disaggregation by two-level demographic groups (eg., age and sex), with covariates*

Description

Used to disaggregate small area population estimates by age, sex, and other socio-demographic or socio-economic characteristics (e.g., ethnicity, religion, educational level, immigration status, etc).

It uses Bayesian hierachical statistical models to predict population proportions and population totals across demographic groups. Primarily designed to support users (e.g., National Statistical Offices) in filling population data gaps across various demographic groups due to outdated or incomplete census/population data.

Usage

```
cheesecake(df, output_dir, verbose = TRUE)
```

Arguments

df	A data frame object containing sample data (often partially observed) on age and sex groups population data, for example, as well as the overall total population counts per administrative unit.
output_dir	This is the directory with the name of the output folder where the disaggregated population proportions and population totals are automatically saved.
verbose	Logical. If TRUE (default), progress messages are displayed during model execution. Set to FALSE to suppress informational messages.

Value

A list of data frame objects of the output files including the disaggregated population proportions and population totals along with the corresponding measures of uncertainties (lower and upper bounds of 95-percent credible intervals) for each demographic characteristic. In addition, a file containing the model performance/model fit evaluation metrics is also produced.

Examples

```
if (requireNamespace("INLA", quietly = TRUE)) {
 data(toydata)
 result <- cheesecake(df = toydata$admin, output_dir = tempdir())
}
```

---

cheesepop *cheesepop: Population disaggregation by two-level demographic groups (eg., age and sex), without covariates*

Description

Similar to the 'cheesecake' function, 'cheesepop' disaggregates small area population estimates by age, sex, and other socio-demographic and socio-economic characteristics (e.g., ethnicity, religion, educational level, immigration status, etc), at the administrative unit level. However, unlike the 'cheesecake' function which uses geospatial covariates to predict missing data values, the 'cheesepop' does not require the use of geospatial covariates.

It uses Bayesian statistical models to predict population proportions and population totals for the demographic groups of interest. Primarily designed to help users in filling population data gaps across demographic groups due to outdated or incomplete census data.

Usage

```
cheesepop(df, output_dir, verbose = TRUE)
```

Arguments

df	A data frame object containing sample data (often partially observed) on age and sex groups population data as well as the estimated overall total counts per administrative unit.
output_dir	This is the directory with the name of the output folder where the disaggregated population proportions and population totals are automatically saved.
verbose	Logical. If TRUE (default), progress messages are displayed during model execution. Set to FALSE to suppress informational messages.

Value

A list of data frame objects of the output files including the disaggregated population proportions and population totals along with the corresponding measures of uncertainties (lower and upper bounds of 95-percent credible intervals) for each demographic characteristic. In addition, a file containing the model performance/model fit evaluation metrics is also produced.

Examples

```
if (requireNamespace("INLA", quietly = TRUE)) {
 data(toydata)
 result <- cheesepop(df = toydata$admin, output_dir = tempdir())
}
```

---

plotHist *plotHist: Produces histogram of the disaggregated population counts across all groups*

Description

This function produces a multi-panel histogram plot of the disaggregated population counts across all the groups. The input data could come from any of the disaggregation functions within the 'jollofR' package (both at admin and grid levels) such as 'cheesecake', 'cheesepop', 'slices', etc.

Usage

```
plotHist(dmat, xlab, ylab)
```

Arguments

dmat	A data frame containing the group-structured disaggregated population estimates which could either be observed or predicted from 'cheesecake', 'cheesepop', 'slices','spices', 'spray' , 'sprinkle', 'splash', 'spray', 'sprinkle1', 'splash1', and 'spray1'.
xlab	A user-defined label for the x-axis (e.g., 'Population Count') considered.
ylab	A user-defined label for the y-axis (e.g., 'Frequency') considered.

Value

A graphic image of histogram of the disaggregated population count

Examples

```
if (requireNamespace("INLA", quietly = TRUE)) {
 data(toydata)
 library(ggplot2)
 result <- cheesepop(df = toydata$admin,output_dir = tempdir())
 plotHist(dmat=result$age_pop,
          xlab="Population Count",
          ylab = "Frequency")
}
```

`plotRast` *plotRast: Produces multi-panel maps of the raster files of the grid-cell disaggregated structured population counts*

Description

This function produces multi-panel maps of the raster files across the various demographic groups of interest. The input data could come from any of the jolloR disaggregation functions at grid cell levels, e.g.,'sprinkle', 'spray', 'splash', 'sprinkle1', 'spray1', 'and splash1',

Usage

```
plotRast(title, output_dir, raster_files, names, nrow, ncol)
```

Arguments

`title`	This is the title of the multi-panel maps of the gridded structured estimates
`output_dir`	The directory for saving the raster files of the disaggregated population estimates
`raster_files`	The names of the raster files to visualize. This must be the same as saved in the raster output folder
`names`	A user-defined names for the plot panels labels. For example, this could be the labels of different age groups. It must be the same length as the 'raster_files'.
`nrow`	Number of rows of the multi-panel maps. The value depends on the number of groups being displayed.
`ncol`	Number of columns of the multi-panel maps. The value depends on the number of groups being displayed. For example, for 12 raster files the products of ncol and nrow must be at least 12.

Value

A graphic image of the multi-panel maps of population disaggregated raster files

Examples

```
if (requireNamespace("INLA", quietly = TRUE)) {
 data(toydata)
 result <- cheesepop(df = toydata$admin,output_dir = tempdir())
 rclass <- paste0("TOY_population_v1_0_age",1:12)
 result2b <- spray(df=result$full_data, rdf=toydata$grid,
                   rclass, output_dir= tempdir())

# make raster maps
        #list.files(output_dir, pattern = "\.tif$",full.names = TRUE) #-
        #use this to see the list of raster files in the directory
group <- 1:12 # customised group
rclass <- paste0("TOY_population_v1_0_age",group)
plt1 <- plotRast(title = "Age disaggregated population counts", # title of the plot
output_dir = tempdir(), # directory where the raster files are saved
raster_files = paste0(output_dir=tempdir(), "/pop_",rclass, ".tif") , # raster files to plot
names = paste0("Age ", group),  # Customised names of the plot panels (same length as rclass)
nrow = 4, ncol =3)# rows and columns of the panels of the output maps
#ggsave(paste0(out_path, "/grid_maps.tif"),#plot = plt1, dpi = 300) - save in output folder
}
```

`pyramid` *pyramid: Produces population pyramid (graphs) of demographics (for cheesecake and cheesepop age-sex output data)*

Description

This function creates population pyramid for age and sex output data from the 'cheesecake' or 'cheesepop' functions outputs. It could also be used to visualize observed age-sex compositions.

Usage

```
pyramid(female_pop, male_pop)
```

Arguments

`female_pop`	A data frame containing the disaggregated population estimates for females across all ages groups.
`male_pop`	A data frame containing the disaggregated population estimates for males across all ages groups.

Value

A graphic image of age-sex population distribution pyramid

Examples

```
if (requireNamespace("INLA", quietly = TRUE)) {
 data(toydata)
 result <- cheesecake(df = toydata$admin, output_dir = tempdir())
 pyramid(result$fem_age_pop,result$male_age_pop)
}
```

`slices` *slices: Disaggregating population counts for a single level of demographics (e.g., age groups only or sex group only) - without covariates Please use 'spices' if you want covariates included.*

Description

This function disaggregates population estimates by a single demographic group (age or sex or religion, etc)

Usage

```
slices(df, output_dir, class, verbose = TRUE)
```

Arguments

`df`	A data frame object containing sample data (often partially observed) on age or sex groups population data as well as the estimated overall total counts per administrative unit.
`output_dir`	This is the directory with the name of the output folder where the disaggregated population proportions and population totals are automatically saved.
`class`	These are the categories of the variables of interest. For example, for educational level, it could be 'no education', 'primary education', 'secondary education', 'tertiary education'.
`verbose`	Logical. If TRUE (default), progress messages are displayed during model execution. Set to FALSE to suppress informational messages.

Value

A list of data frame objects of the output files including the disaggregated population proportions and population totals along with the corresponding measures of uncertainties (lower and upper bounds of 95-percent credible intervals) for each demographic characteristic. In addition, a file containing the model performance/model fit evaluation metrics is also produced.

Examples

```
if (requireNamespace("INLA", quietly = TRUE)) {
 data(toydata)
 library(dplyr)
 classes <- names(toydata$admin %>% dplyr::select(starts_with("age_")))
 result2 <- slices(df = toydata$admin, output_dir = tempdir(), class = classes)
}
```

`spices` *spices: Disaggregates population counts for a single level of demographics (e.g., age groups only or sex group only) with covariates.*

Description

This function disaggregates population estimates by a single demographic (age or sex or religion, etc)

Usage

```
spices(df, output_dir, class, verbose = TRUE)
```

Arguments

`df`	A data frame object containing sample data (often partially observed) on age or sex groups population data as well as the estimated overall total counts per administrative unit.
`output_dir`	This is the directory with the name of the output folder where the disaggregated population proportions and population totals are automatically saved.
`class`	This are the categories of the variables of interest. For example, for educational level, it could be 'no education', 'primary education', 'secondary education', 'tertiary education'.
`verbose`	Logical. If TRUE (default), progress messages are displayed during model execution. Set to FALSE to suppress informational messages.

Value

A list of data frame objects of the output files including the disaggregated population proportions and population totals along with the corresponding measures of uncertainties (lower and upper bounds of 95-percent credible intervals) for each demographic characteristic. In addition, a file containing the model performance/model fit evaluation metrics is also produced.

Examples

```
if (requireNamespace("INLA", quietly = TRUE)) {
 data(toydata)
 library(dplyr)
 classes <- names(toydata$admin %>% dplyr::select(starts_with("age_")))
 result2 <- spices(df = toydata$admin, output_dir = tempdir(), class = classes)
}
```

`splash` *splash: Disaggregates population counts at high-resolution grid cells using building counts values of grid cells as a weighting layer. It is used for two-level disaggregation (e.g., age and sex).*

Description

This function disaggregates population estimates at grid cell levels using the building counts of each grid cell to first disaggregate the admin unit's total population across the grid cells. Then, each grid cell's total count is further disaggregated into groups of interest using the admin's proportions.

Usage

```
splash(df, rdf, rclass, output_dir, verbose = TRUE)
```

Arguments

`df`	A data frame object containing sample data (often partially observed) on different demographic groups population. It contains the admin's total populatioin count to be disaggregated as well as other key variables as defined within the 'toydata'.
`rdf`	A gridded data frame object containing key information on the grid cells. Variables include the admin_id which must be identical to the one in the admin level data. It contains GPS coordinates. i.e, longitude (lon) and Latitude (lat) of the grid cell's centroids.
`rclass`	This is a user-defined names of the files to be saved in the output folder.
`output_dir`	This is the directory with the name of the output folder where the disaggregated population proportions and population totals are automatically saved.
`verbose`	Logical. If TRUE, prints progress messages. Default is TRUE.

Value

A list of data frame objects of the output files including the disaggregated population proportions and population totals along with the corresponding measures of uncertainties (lower and upper bounds of 95-percent credible intervals) for each demographic characteristic. In addition, a file containing the model performance/model fit evaluation metrics is also produced.

Examples

```
if (requireNamespace("INLA", quietly = TRUE)) {
 # load key libraries
 library(raster)
 library(dplyr)
 library(terra)
 # load toy data
 data(toydata)
 # run 'cheesepop' to obtain admin-level proportions
 result <- cheesepop(df = toydata$admin,output_dir = tempdir())
 # specify the names to assign to the raster files
 rclass <- paste0("TOY_population_v1_0_age",1:12)
 # run the splash function to disaggregate at grid cells
 result2 <- splash(df = result$full_data, rdf = toydata$grid, rclass, output_dir = tempdir())
 # read and visualise one of the saved raster files
 ras2<- rast(paste0(output_dir = tempdir(), "/pop_TOY_population_v1_0_age4.tif"))
 plot(ras2)
}
```

`splash1` *splash1: Disaggregates population counts at high-resolution grid cells using building counts values of grid cells as a weighting layer. However, unlike 'splash' it is used for one-level disaggregation*

Description

This function disaggregates population estimates at grid cell levels for one level of classification only. It uses the building counts of each grid cell to first disaggregate the admin unit's total population across the grid cells. Then, each grid cell's total count is further disaggregated into groups of interest using the admin's proportions.

Usage

```
splash1(df, rdf, class, rclass, output_dir, verbose = TRUE)
```

Arguments

`df`	A data frame object containing sample data (often partially observed) on different demographic groups population. It contains the admin's total populatioin count to be disaggregated as well as other key variables as defined within the 'toydata'.
`rdf`	A gridded data frame object containing key information on the grid cells. Variables include the admin_id which must be identical to the one in the admin level data. It contains GPS coordinates. i.e, longitude (lon) and Latitude (lat) of the grid cell's centroids.

class	These are the categories of the variables of interest. For example, for educational level, it could be 'no education', 'primary education', 'secondary education', 'tertiary education'.
rclass	This is a user-defined names of the files to be saved in the output folder.
output_dir	This is the directory with the name of the output folder where the disaggregated population proportions and population totals are automatically saved.
verbose	Logical. If TRUE, prints progress messages. Default is TRUE.

Value

A list of data frame objects of the output files including the disaggregated population proportions and population totals along with the corresponding measures of uncertainties (lower and upper bounds of 95-percent credible intervals) for each demographic characteristic. In addition, a file containing the model performance/model fit evaluation metrics is also produced.

Examples

```
if (requireNamespace("INLA", quietly = TRUE)) {
 # load key libraries
 library(raster)
 library(dplyr)
 library(terra)
 # load toy data
 data(toydata)
 # run 'cheesepop' to obtain admin-level proportions
 result <- cheesepop(df = toydata$admin,output_dir = tempdir())
 # specify the names to assign to the raster files
 class <- names(toydata$admin %>% dplyr::select(starts_with("age_")))
 rclass <- paste0("TOY_population_v1_0_age",1:12)
 # run the splash function to disaggregate at grid cells
 result2 <- splash1(df = result$full_data, rdf = toydata$grid,
 class, rclass, output_dir = tempdir())
 # read and visualise one of the saved raster files
 ras2<- rast(paste0(output_dir = tempdir(), "/pop_TOY_population_v1_0_age4.tif"))
 plot(ras2)
}
```

---

spray *Spray: Disaggregates population counts by dividing the admin total by the number of grid cells within the administrative units. Then admin proportions are used to further disaggregate the grid cell totals by groups*

Description

This function disaggregates population estimates at grid cell levels when there are no information on the building and population counts.

Usage

```
spray(df, rdf, rclass, output_dir, verbose = TRUE)
```

Arguments

df	A data frame object containing sample data (often partially observed) on different demographic groups population. It contains the admin's total populatioin count to be disaggregated as well as other key variables as defined within the 'toydata'.
rdf	A gridded data frame object containing key information on the grid cells. Variables include the admin_id which must be identical to the one in the admin level data. It contains GPS coordinates. i.e, longitude (lon) and Latitude (lat) of the grid cell's centroids.
rclass	This is a user-defined names of the files to be saved in the output folder.
output_dir	This is the directory with the name of the output folder where the disaggregated population proportions and population totals are automatically saved.
verbose	Logical. If TRUE, prints progress messages. Default is TRUE.

Value

A list of data frame objects of the output files including the disaggregated population proportions and population totals along with the corresponding measures of uncertainties (lower and upper bounds of 95-percent credible intervals) for each demographic characteristic. In addition, a file containing the model performance/model fit evaluation metrics is also produced.

Examples

```
if (requireNamespace("INLA", quietly = TRUE)) {
 # load relevant libraries
 library(raster)
 library(terra)
 # load toy data
 data(toydata)
 # run 'cheesepop' function for admin level disaggregation
 result <- cheesepop(df = toydata$admin,output_dir = tempdir())
 rclass <- paste0("TOY_population_v1_0_age",1:12) # Mean
 # run 'spray' for grid cell level disaggregation
result2 <- spray(df = result$full_data, rdf = toydata$grid, rclass, output_dir = tempdir())
 ras2<- rast(paste0(output_dir = tempdir(), "/pop_TOY_population_v1_0_age4.tif"))
 plot(ras2) # visualize
}
```

spray1 *spray1: Disaggregates population counts at high-resolution grid cells in the absence population and building counts - for one-level only*

Description

This function disaggregates population estimates at grid cell levels using the building counts of each grid cell to first disaggregate the admin unit's total population across the grid cells. Then, each grid cell's total count is further disaggregated into groups of interest using the admin's proportions.

Usage

```
spray1(df, rdf, class, rclass, output_dir, verbose = TRUE)
```

Arguments

df	A data frame object containing sample data (often partially observed) on different demographic groups population. It contains the admin's total populatioin count to be disaggregated as well as other key variables as defined within the 'toydata'.
rdf	A gridded data frame object containing key information on the grid cells. Variables include the admin_id which must be identical to the one in the admin level data. It contains GPS coordinates. i.e, longitude (lon) and Latitude (lat) of the grid cell's centroids.
class	These are the categories of the variables of interest. For example, for educational level, it could be 'no education', 'primary education', 'secondary education', 'tertiary education'.
rclass	This is a user-defined names of the files to be saved in the output folder.
output_dir	This is the directory with the name of the output folder where the disaggregated population proportions and population totals are automatically saved.
verbose	Logical. If TRUE, prints progress messages. Default is TRUE.

Value

A list of data frame objects of the output files including the disaggregated population proportions and population totals along with the corresponding measures of uncertainties (lower and upper bounds of 95-percent credible intervals) for each demographic characteristic. In addition, a file containing the model performance/model fit evaluation metrics is also produced.

Examples

```
if (requireNamespace("INLA", quietly = TRUE)) {
 library(raster) # load relevant libraries
 library(dplyr)
 library(terra)
 data(toydata) # load toy data
```

```
 # run 'cheesepop' admin unit disaggregation function
 result <- cheesepop(df = toydata$admin,output_dir = tempdir())
 class <- class <- names(toydata$admin %>% dplyr::select(starts_with("age_")))
 rclass <- paste0("TOY_population_v1_0_age",1:12)

 # run spray1 grid cell disaggregation function
result2 <- spray1(df = result$full_data, rdf = toydata$grid, class, rclass, output_dir = tempdir())
 ras2<- rast(paste0(output_dir = tempdir(), "/pop_TOY_population_v1_0_age4.tif"))
 plot(ras2) # visulize of the raster files produced
}
```

`sprinkle` *sprinkle: Disaggregates population counts at high-resolution grid cells using the grid cell's total population counts. Note that this could also be applied to more than two levels scenarios*

Description

This function disaggregates population estimates at grid cell levels using the population counts of each grid cell.

Usage

```
sprinkle(df, rdf, rclass, toSave, rasterToCSV, output_dir, verbose = TRUE)
```

Arguments

`df`	A data frame object containing sample data (often partially observed) on different demographic groups population. It contains the admin's total populatioin count to be disaggregated as well as other key variables as defined within the 'toydata'.
`rdf`	A gridded data frame object containing key information on the grid cells. Variables include the admin_id which must be identical to the one in the admin level data. It contains GPS coordinates. i.e, longitude (lon) and Latitude (lat) of the grid cell's centroids.
`rclass`	This is a user-defined names of the files to be saved in the output folder.
`toSave`	Specifies the raster files to save - it has three options: set toSave="prop" to write only age and age-sex proportion files only, or set toSave = "pop" to write age and age-sex population count files only, or set toSave = "everything" to write everything including lower and upper bounds of 95-percent credible interval.
`rasterToCSV`	This is used to declare whether the raster files should also be saved as .CSV files: set rasterToCSV = NULL to skip, or set rasterToCSV = TRUE to write the corresponding .CSV files. Note that the length of time taken depends on the size of the files.

output_dir	This is the directory with the name of the output folder where the disaggregated population proportions and population totals are automatically saved.
verbose	Logical. If TRUE, prints progress messages. Default is TRUE.

Value

A list of data frame objects of the output files including the disaggregated population proportions and population totals along with the corresponding measures of uncertainties (lower and upper bounds of 95-percent credible intervals) for each demographic characteristic. In addition, a file containing the model performance/model fit evaluation metrics is also produced.

Examples

```
if (requireNamespace("INLA", quietly = TRUE)) {
 # load necessary libraries
 library(raster)
 library(terra)
 # load toy data
 data(toydata)
 # run 'cheesepop' function for admin level disaggregation
 result <- cheesepop(df = toydata$admin,output_dir = tempdir())
 rclass <- paste0("TOY_population_v1_0_age",1:12)
 # run 'sprinkle' function for grid cell disaggregation and save
 result2 <-  sprinkle(df = result$full_data, rdf = toydata$grid, rclass,
 toSave="pop",rasterToCSV = NULL,  output_dir = tempdir())
 ras2<- rast(paste0(tempdir(), "/pop_TOY_population_v1_0_age4.tif"))
 plot(ras2) # visulize raster
}
```

---

sprinkle1 *sprinkle1: Disaggregates population counts at high-resolution grid cells using the grid's total population. This can also be applied to one-level disaggregation*

Description

This function disaggregates population estimates at grid cell levels using the population counts of each grid cell.

Usage

```
sprinkle1(df, rdf, class, rclass, output_dir, verbose = TRUE)
```

Arguments

df	A data frame object containing sample data (often partially observed) on different demographic groups population. It contains the admin's total populatioin count to be disaggregated as well as other key variables as defined within the 'toydata'.
rdf	A gridded data frame object containing key information on the grid cells. Variables include the admin_id which must be identical to the one in the admin level data. It contains GPS coordinates. i.e, longitude (lon) and Latitude (lat) of the grid cell's centroids.
class	These are the categories of the variables of interest. For example, for educational level, it could be 'no education', ' primary education', 'secondary education', 'tertiary education'.
rclass	This is a user-defined names of the files to be saved in the output folder.
output_dir	This is the directory with the name of the output folder where the disaggregated population proportions and population totals are automatically saved.
verbose	Logical. If TRUE, prints progress messages. Default is TRUE.

Value

A list of data frame objects of the output files including the disaggregated population proportions and population totals along with the corresponding measures of uncertainties (lower and upper bounds of 95-percent credible intervals) for each demographic characteristic. In addition, a file containing the model performance/model fit evaluation metrics is also produced.

Examples

```
if (requireNamespace("INLA", quietly = TRUE)) {
 # load relevant libraries
 library(raster)
 library(dplyr)
 library(terra)
 # load the toy data
 data(toydata)
 # run 'cheesepop' function for admin level disaggregation
 result <- cheesepop(df = toydata$admin,output_dir = tempdir())
 class <- names(toydata$admin %>% dplyr::select(starts_with("age_")))

 rclass <- paste0("TOY_population_v1_0_age",1:12)
 # run 'sprinkle1' function for grid cell disaggregation at one level
 result2 <- sprinkle1(df = result$full_data,
 rdf = toydata$grid, class, rclass, output_dir = tempdir())
 ras2<- rast(paste0(output_dir = tempdir(), "/pop_TOY_population_v1_0_age4.tif"))
 plot(ras2) # visulize raster
}
```

`toydata`	*A list object containing two dataframes - an administrative-level dataset (admin) containing partially observed age-sex structured data, and a grid-cell level dataset (grid) for population disaggregation at 1km by 1km grid cells.*

Description

Artificially generated toy datasets that come in a cross-sectional format. The 'admin' data is a dataframe collated at administrative unit level which contains information on the observed number of individuals per age and sex groups within each administrative unit. Key variables include the administrative unit identifier (admin_id), the admin total population to be disaggregated (total), the total number of buildings within each admin unit (bld), and the longitude (lon) and latitude (lat). The 'admin' data provides artificial information for 900 spatially distinct administrative units in which the individuals in the population are grouped into 12 mutually exclusive and exhaustive age groups. Each of the age groups was further grouped into 'male' and 'female' groups. The data contains the total population counts (total) for each spatial unit but also contains missing age and sex groups population counts. The model first predicts the population proportions of the missing data and then disaggregates the population totals using the predicted proportions to obtain the predicted population counts for the age and sex groups. Note that the same applies for other demographic groups such as marital status, race, etc.

Usage

```
data(toydata)
```

Format

An object of class `"list"`

admin_id Available in both the 'admin' and 'grid' datasets. It is a numerical value which serves as the administrative units unique identifier. They should match perfectly for both the 'admin' and grid' datasets

grd_id Available in the 'grid' dataset only. It is a numerical value which serves as the grid cell unique indentifier.

x1,x2,x3 These are the samples of geospatial covariates (only required for the 'cheesecake' and the 'slices' functions). Note that these are the covariates identified to significantly predict population distribution among the demographic groups. The package allows the user to include any number of covariates in their own datasets.

total Available in both the 'admin' and 'grid' datasets. It provides estimates of the total population counts to be disaggregated. It DOES NOT necessarily have to be a rowsum of the age groups totals.

bld Available in both the 'admin' and 'grid' datasets. It provides the total number of buildings in each grid cell or administrative unit.

set_typ Administrative unit's settlement type classification (e.g., urban, rural).

edu_no, edu_prim, edu_sec, edu_high These are the fully or partially observed number of people by the highest educational level of the household members. Here, edu_no = no education, edu_prim = primary education, edu_sec = secondary education, and edu_high = higher education.

age_1, ..., age_12 These correspond to the partially or fully observed number of people for each age group. Note that only 12 age groups are used here for illustration purposes,however, the package can accommodate any number of age or sex or any demographic groups.

fage_1, ..., fage_12 These correspond to the partially or fully observed number of females corresponding to each of the age groups. Note that only 12 age groups are used here for illustration purposes,however, the package can accommodate any number of age or sex or any demographic groups.

mage_1, ..., mage_12 These correspond to the partially or fully observed number of males corresponding to each of the age groups. Note that only 12 age groups are used here for illustration purposes,however, the package can accommodate any number of age or sex or any demographic groups.

lon Available in both the 'admin' and 'grid' datasets. Provides the value of the longitude of the centroids of the grid cells or admin unit polygons.

lat Available in both the 'admin' and 'grid' datasets. Provides the value of the latitude of the centroids of the grid cells or admin unit polygons.

Details

The second dataset in the toydata list is the 'grid' data which allows for the prediction of the age-sex structures at 1km by 1km grid cells (note that population predictions can be made at any spatial resolution of interest). The 'grid' data contains six key variables. These are administrative unit identifier (admin_id) which must be identical to the those in the 'admin' data; the grid cell identifier (grd_id); the total number of people per grid cell (total), if available; the total number of buildings per grid cell (bld), if available; and the longitude (lon) and latitude (lat) variables for the grid cell centroids.

illustrate the use of the package.

References

This data set was artificially created for the purpose of illustrations within the jollofR package.

Examples

```
data(toydata)
head(toydata$admin)
head(toydata$grid)
```

Index

∗ **datasets**
toydata, 18

boxLine, 2

cheesecake, 3
cheesepop, 4

plotHist, 5
plotRast, 6
pyramid, 7

slices, 8
spices, 9
splash, 10
splash1, 11
spray, 12
spray1, 14
sprinkle, 15
sprinkle1, 16

toydata, 18